AD NUMBER

ADB010612

LIMITATION CHANGES

TO:

Approved for public release; distribution is unlimited.

FROM:

Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; 30 JUN 1975. Other requests shall be referred to Air Force Avionics Laboratory, Wright-Patterson AFB, OH 45433.

AUTHORITY

AFAL ltr, 12 Sep 1977

AFAL-TR-75-150

CSEL BIT ERROR RATE COUNTERS

COMPUTER SCIENCES CORPORATION
6565 Arlington Boulevard
Falls Church, Virginia 22046

February 1976

TECHNICAL REPORT AFAL-TR-75-150

Final Report for Period 1 March 1974 – 1 May 1975

AIR FORCE AVIONICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

Page 58 left blank intentionally
per Mrs Dolen.

J.L.C.
4/30/76

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This final report was submitted by Computer Sciences Corporation, 6565 Arlington Blvd., Falls Church, Virginia 22046, under contract F33615-74-C-1155, job order 12271215, with Air Force Avionics Laboratory, Wright-Patterson AFB, Ohio. Wade T. Hunt AFAL/AAF-2 was the Laboratory Project Engineer.

This technical report has been reviewed and is approved for publication.

Wade T. Hunt
WADE T. HUNT
Project Engineer/Scientist

Harry L Deal
HARRY L. DEAL
Supervisor

FOR THE COMMANDER

George F. Cudahy
GEORGE F. CUDAHY, Col, USAF
Chief, System Avionics Division
AF Avionics Laboratory

AIR FORCE - 22 MARCH 1976 - 50

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFAL-TR-75-150	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): CSEL Bit Error Rate Counters.	5. TYPE OF REPORT & PERIOD COVERED: Final Technical Report. 1 March 1974-1 May 1975
6. PERFORMING ORG. REPORT NUMBER: CSC-3314-000000	
7. AUTHOR(s): Robert A. Glicksman, Charles F. Pavey, Frederick J. Rose	8. CONTRACT OR GRANT NUMBER(s): F33615-74-C-1155
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Computer Sciences Corporation, 6565 Arlington Boulevard, Falls Church, VA 22046	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Project 1227, Task 1227 12
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Avionics Laboratory, Air Force Wright Aeronautical Laboratories, Air Force Systems Command, WPAFB, Ohio 45433	12. REPORT DATE: February 1976
	13. NUMBER OF PAGES: 103
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution Limited to U.S. Government only;
Reason, test and evaluation; 30 June 1975
Other requests for this document must be referred to AFAL/AAF-2

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer, software, error rate, counters, programming, hardware, data generator, subroutines.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the addition and integration of two Bit Error Rate Counter (BERC) units into the K-Band Terminal Simulator portion of the AFAL Communication Systems Evaluation Laboratory (CSEL). These two units will provide the capability to measure bit error rate on two classes of signals. The first is represented by single channel, low speed (75 to 9600 bits/second),

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

405 717

20. Abstract (Cont'd)

asynchronous or synchronous data. The second are high speed (to 10 megabits/second) synchronous data, usually generated by time domain multiplexing equipment for wideband communications systems. The original software for the CSEL has been modified to allow operator interaction with the BERCs. This interaction is achieved in a manner consistent with all the other elements of CSEL.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

This Page Intentionally Left Blank

SECTION I

INTRODUCTION

This report deseribes the addition and integration of two Bit Error Rate Counter (BERC) units into the K-Band Terminal Simulator portion of the AFAL Communication Systems Evaluation Laboratory (CSEL). These two units will provide a capability for two classes of signals. The first, International Data Sciences Corporation Model 1210A, is represented by a single channel, low speed (75 to 9600 bps), asynchronous or synchronous data. The second, International Data Sciences Corporation Model 3000, is a high speed (to 10 Mbps) synchronous data unit, usually generated by time domain multiplexing equipment for wideband communications systems. These units have been modified to allow automatic computer control of all of their functions. In addition, they can be operated locally on a standalone basis or remotely under control of the CSEL software.

The software involves additions and modifications to the CSEL software package on three levels. On the lowest level, closest to the hardware interfaces, are software drivers. These provide a vehicle by which other software can communicate with the BERCs. Higher up the software hierarchy are procedures. The CSEL software consists of structures such that each type of device has a procedure or a set of procedures that control the flow of information to and from that device. The highest level of software is the System. Here, additional user's parameters are incorporated into the system structure to allow the operator to form bit error rate tests within the static mode of operation.

SECTION II

OPERATIONS

1. GENERAL

Figure 1 shows the physical arrangement of the BERCs, patch panel, and monitor panel. The patch panel (Figure 2) enables the user to connect the BERCs to the system under test from either the front of the rack or via the I/O panel at the rear. Figure 1 shows the BERCs connected to the rear I/O panel via the patch panel. Figure 3 shows the routing of the data and timing cabling in the cabinet. The monitor panel (Figure 4) is divided into two independent sections; one for the 1210A and one for the 3000. Each section contains a switch for selection of either the REMOTE (BERC controlled by CSEL software) or MANUAL (BERC operated locally, independent of the CSEL). Each section also contains indicator lamps which give a visual display of the following parameters.

1. ERROR COUNT denotes the number of errors which have occurred in a selected test sequence. This count is displayed in binary (bit Ø is the LSB) and reflects the state of the error counter in the interface logic.

2. COUNT OVERFLOW indicates that the error counter in the interface logic has overflowed at least once (see Figure 4). This indicator has no relation to the one on the respective BERC because the error counters in each BERC overflow after a count of 999 whereas those in each interface overflow after a count of 4095.

3. IN SYNC, when lit, indicates that the receiver is locked to the incoming data.

4. SYNC RECOVERED, when lit, indicates that the receiver has lost sync and recovered it at least once.

5. READY, when lit, indicates that the BERC has finished a test. When the BERC is operating in the remote mode, READY indicates that the BERC is ready to accept new instructions from the CSEL system.

2. MANUAL MODE

To operate either BERC in the manual mode, place the associated REMOTE/MANUAL switch in the MANUAL position (Figure 4). With the switch in this position, the BERC is operable from its front panel controls and will ignore inputs from the CSEL computer. For operating instructions of either BERC in the manual mode, reference should be made to the appropriate manufacturer's instruction manual.

3. REMOTE MODE

To operate either BERC in the remote mode, place the associated REMOTE/MANUAL switch in the REMOTE position (Figure 4). With the switch in this position, the BERC is operable only from the CSEL computer. The only front panel control which is operative in the remote mode is the TEST/OPERATE switch on the front panel of the 1210A. This switch is functional in both the manual and remote modes. (See the 1210A instruction manual for the function of this switch.) Section IV describes the remote mode of operation in detail.

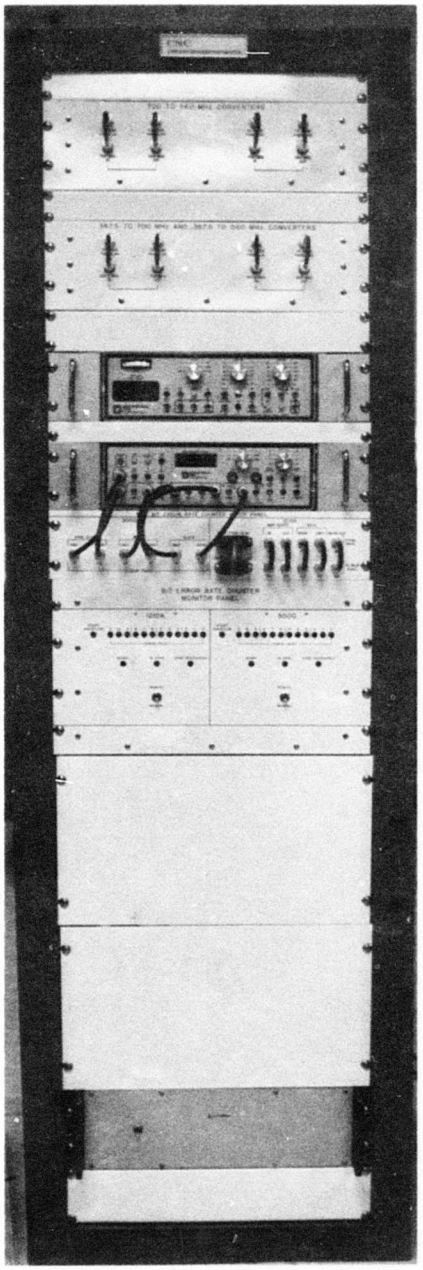


Figure 1. BERCs, Patch Panel, and Monitor Panel

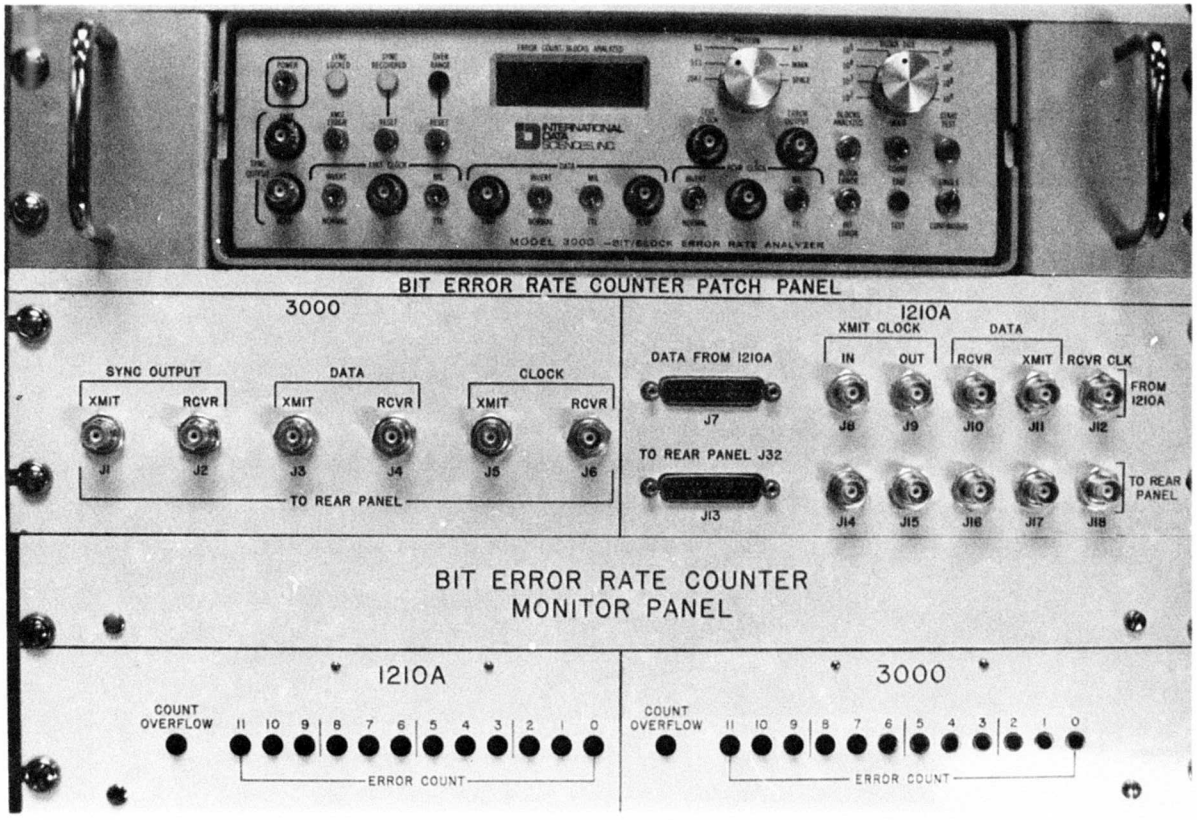


Figure 2. Patch Panel

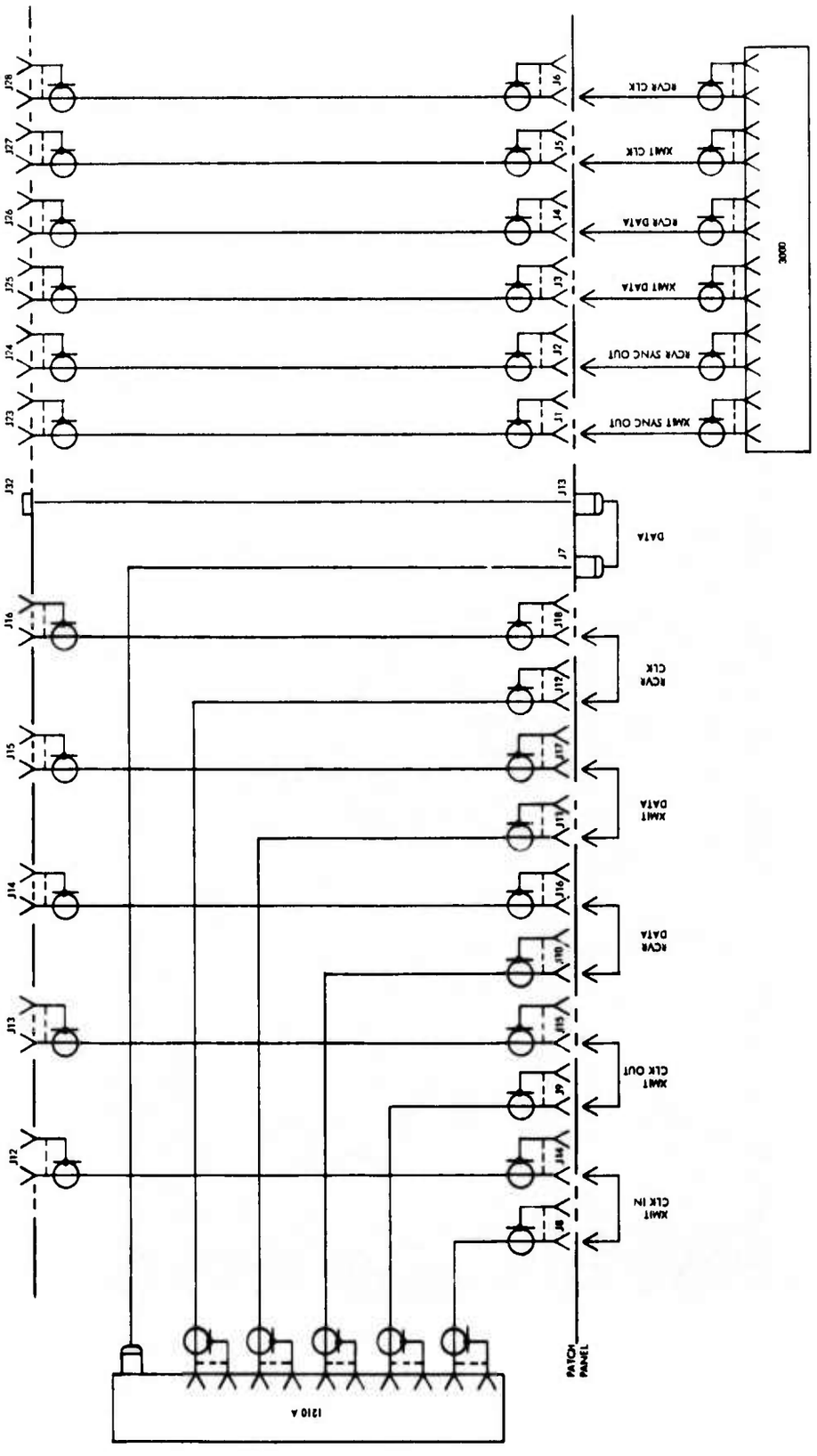


Figure 3. Data and Timing Cabling, Block Diagram

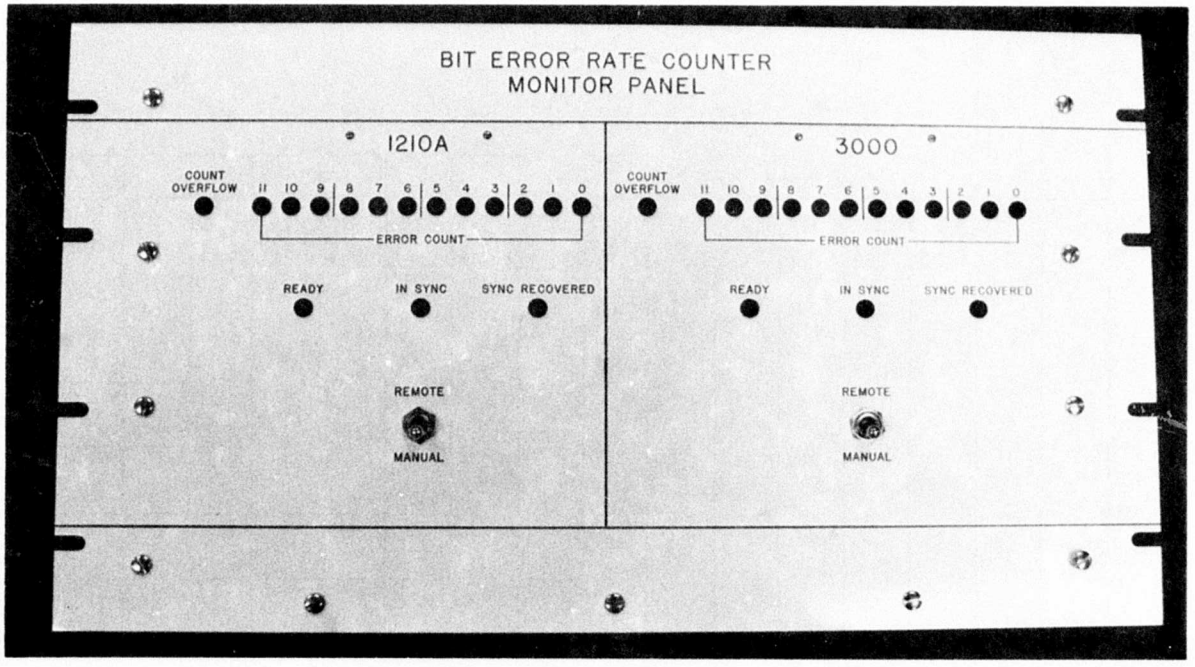


Figure 4. Monitor Panel

SECTION III

INTERFACE DESCRIPTION

1. GENERAL

A block diagram of the interface system for either the Model 1210A or 3000 is shown in Figure 8. Although the block diagram is the same for both models, there are some differences in the individual interfaces. The principal difference is that the Model 3000 requires fewer data generator control signals, so the interface controlling this model is somewhat simpler.

As shown in the figure, all signals that control the BERC data formats and rates are interfaced to the PDP-11 computer through the UDC-11. The data generator control interface retransmits these control signals to input registers in the UDC-11 for verification. All signals that control the measurement functions and all measurement status signals and data are interfaced to the PDP-11 computer through a DR11-A general purpose interface.

Certain BERC programming control signals from the UDC-11 and DR11-A control registers are encoded within functionally consistent groupings of bits and then decoded externally within the BERC controller. These decoded bit groupings are used to drive the control logic in the modified BERC. Delay for the response time of the interfacing logic and BERC is provided by the software (where the signals come through the UDC-11) and by the interface hardware (where the signals come through the DR11-A command register).

2. UDC 11 INTERFACE FUNCTIONS

As previously mentioned, the signals which control data format (in the Model 3000) or data rate and data format (in the Model 1210A) are interfaced to the computer through the UDC-11.

Figure 5 shows the bit assignments of the UDC-11 control and status registers. Table 1 explains the functions of the bit groupings in these registers.

SECTION	FUNCTION	ADDRESS	DATA FORMAT (MSB → LSB)															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BERC Data Generator	1210A Data Generator Control "A"	771160	•	RTS ON	DTR ON	SYNC / ASYNC	2 XMIT Data	1 XMIT Data	2 Rcv Data	1 Rcv Data	2 XMIT Clk	1 XMIT Clk	2 Rcv Clk	1 Rcv Clk	XMIT Error	4 Pattern	2 Pattern	1 Pattern
	1210A Data Generator Control "B"	771162	•	•	•	•	•	•	•	•	•	•	•	•	8	4 Data Rate	2 Data Rate	1 Data Rate
	1210A Data Generator Input Select	771164	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	Select A / B
	1210A Data Generator Input	771044	For Bit 0 of 771164 = 0, Same As 771160; For Bit 0 of 771164 = 1, Same As 771162															
	3000 Data Generator Control	771166	•	•	•	•	•	•	Inv / Norm Data	MIL / TTL Data	Inv / Norm XMIT Clk	MIL / TTL XMIT Clk	Inv / Norm Rcv Clk	MIL / TTL Rcv Clk	XMIT Error	4 Pattern	2 Pattern	1 Pattern
	3000 Data Generator Input	771046	← Same As 771166 →															

Figure 5. UDC-11 Register Bit Assignments

TABLE 1

UDC-11 REGISTER BIT FUNCTIONS

Register	Signal	Bits	Description
1210A DATA GENERATOR	PATTERN	0-2	Contains, in coded form, data sequence utilized in 1210A in a given test (see Table 5-1)
	XMIT ERROR	3	When a "1", causes the 1210A to insert one error in each block of data analyzed
	RCV CLK REV DATA	4,5 8,9	Control voltage levels which receiver clock and data inputs are responsive to (see 1210A Instruction Manual)
	XMIT CLK XMIT DATA	6,7 10,11	Control voltage levels of transmitter clock and data outputs (see 1210A Instruction Manual)
	SYNC/ASYNC	12	When "1", causes 1210A to operate in synchronous mode; when "0" causes 1210A to operate in asynchronous mode
	DTR ON	13	When "1", causes 1210A to assert DTR line on its data connector (see 1210A Instruction Manual)
	RTS ON	14	When "1", causes 1210A to assert RTS line on connector (see 1210A Instruction Manual)
		15	Not used
1210A DATA GENERATOR CONTROL B	DATA RATE	0-3	Contain, in coded form, rate of transmitted data (see Table 5-2)
		4-15	Not used
1210A DATA GENERATOR INPUT SELECT	SELECT A/B	1	Determines whether control signals from either 1210A CONTROL A OR CONTROL B register are retransmitted from 1210A data generator control interface logic to 1210A Data Generator Input status register. A "zero" selects A and a "one" selects B.
		2-15	Not used
1210A DATA GENERATOR INPUT		0-14	Accepts status inputs as determined by select A/B bit
		15	Not used
3000 DATA GENERATOR CONTROL	PATTERN	0-2	Contains in coded form, data sequence utilized in 3000 in a given test (see Table 5-3)
	XMIT ERROR	3	Causes 3000 to insert one error in test data each time bit 3 goes from "0" to a "1". Unlike the XMIT ERROR control for the 1210A, which is a static level control, this signal is dynamic in nature.
	RCVR CLK XMIT CLK DATA	4,5 6,7 8,9	Control voltage levels which receiver clock, transmit clock and data inputs are responsive to (see 3000 Instruction Manual)
		10-15	Not used
3000 DATA GENERATOR		0-9	Receives 3000 Data Generator Control bits retransmitted from 3000 data generator control logic for verification.
		10-15	Not used

In preparing for a test, the UDC-11 control registers are loaded with the bit groupings pertaining to the selected data rate and/or format. These bit groupings are sent to the data generator portion of the BERC interface logic where they are decoded as necessary, buffered and sent to the BERC

3. DR11-A INTERFACE FUNCTION

All signals pertaining to the data measurement functions of the BERC are interfaced to the PDP-11 computer through the DR11-A general purpose interface.

Figure 6 shows the bit assignments of the DR11-A registers and Table 6 explains their functions. When an actual measurement is to be made, the DR11-A control and status register is first checked to ensure that the BERC is in the remote mode. The READY flag may or may not be set since it is of no concern at this time and is not tested.

If remote mode is indicated by the MAN/REMOTE flag, the DR11-A command word register is loaded. This action causes the control and status word register READY flag to go "busy." The command register now contains the bit groupings as listed in Figure 6.

The block size is decoded in the interface logic hardware and sent on to the BERC. The SYNC RECOVERED indicator (in the interface logic only) may also be reset at this time, if command word bit 06 is a "one". The error counters and the OVERFLOW indicators in both the BERC and controller hardware are also reset if the RESET COUNT AT START bit is on. This bit allows a fresh reading of errors for each test or a totalization of errors over a desired number of tests.

The state of the BIT/BLOCK ERROR bit determines whether bit or block errors are counted and the state of the SINGLE/CONTINUOUS bit determines whether a single or continuous test is performed. These bits are normally set for bit error and single block tests.

SECTION	FUNCTION	ADDRESS	DATA FORMAT (MSB 15 … LSB 0)
1210A Meter Controller	1210A Control & Status Word	Vector A: 360 Vector B: 364 767730	15: MAN/Remote Flag; 14–8: •; 7: Ready Flag; 6: Ready Int Enab; 5: MAN Int Enab; 4–0: •
	1210A Command Word	767732	15–9: •; 8: Start Test; 7: Reset SYNC Recovered; 6: Reset Count At Start; 5: Single Block Cont.; 4: Bit/Block Error; 3–0: Block Size (8, 4, 2, 1)
	1210A Data Word	767734	15: •; 14: SYNC Recovered; 13: In/Out of SYNC; 12: Count Over-Flow; 11–0: Binary Data (11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0)
3000 Meter Controller	3000 Control & Status Word	Vector A: 370 Vector B: 374 767720	15: MAN/Remote Flag; 14–8: •; 7: Ready Flag; 6: Ready Int Enab; 5: MAN Int Enab; 4–0: •
	3000 Command Word	767722	15–9: •; 8: Start Test; 7: Reset SYNC Recovered; 6: Reset Count At Start; 5: Single Block Cont; 4: Bit/Block Error; 3–0: Block Size (8, 4, 2, 1)
	3000 Data Word	767724	15: •; 14: SYNC Recovered; 13: In/Out of SYNC; 12: Count Over-Flow; 11–0: Binary Data (11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0)

Figure 6. DR11-A Register Bit Assignments

TABLE 2

DR11-A REGISTER BIT FUNCTIONS

Register	Signal	Bits	Description
CONTROL & STATUS WORD	MAN. INT. ENABLE	5	A eontrol from PDP-11 CPU; determines whether MAN/REMOTE flag is treated as a status input or as an interrupt.
	READY INT ENABLE	6	A control from PDP-11 CPU; determines whether READY flag is treated as a status input or as an interrupt.
	READY FLAG	7	A status input from meter control interfaee logic; when "1", interface logic is ready to accept next instruetion from CSEL.
	MAN/REMOTE	15	A status input from meter control interface logic; indicates position of MAN/REMOTE switch on bit error rate counter monitor panel. "O" indicates remote, "1" indicates manual.
DATA WORD	BINARY DATA	0-11	Contains binary error count accumulated in meter control interface 12-bit counter.
	COUNT OVERFLOW	12	A "1" indieates that meter control interface error eounter has overflowed at least once.
	IN/OUT OF SYNC	13	A "1" indicates that data generator and receiver sections of BERC are in syne; a "0" indicates loss of sync. This is a real-time signal and has no memory.
	SYNC RECOVERED	14	A "1" indicates that BERC has detected a loss of sync and has subsequently recovered syne at least once.
COMMAND WORD	BLOCK SIZE	0-3	Contains, in coded form, number of bits or bloeks of bits analyzed in a given test (see Table 5-4 for 1210A; Table 5-5 for 3000)
	BIT/BLOCK ERROR	4	Determines whether error analysis is done on a bit-by-bit basis or by comparing a block of transmitted and received data (sce BERC Instruction Manual) "1" = BIT ERROR "0" = BLOCK ERROR
	SINGLE BLOCK	5	Determines whether predetermined number of bits or block of data are analyzed as determined by BLOCK SIZE bits, or whether data analysis is done on a continuous basis. "1" = SINGLE "0" = CONTINUOUS
	RESET COUNT AT START	6	A "1" causes error counters and OVERFLOW indicators in both BERC and interface logic to be reset prior to start of test.
	RESET SYNC RECOVERED	7	A "1" causes SYNC RECOVERED indicator in interface logle only to be reset prior to start of test.
	START TEST	8	A "1" causes BERC to start next test sequence.
		9-15	Not used

The meter control interface logic "times out" the performance of these functions and then issues a "start test" command to the BERC. After a block of data, as programmed in the command word, has been run by the BERC, it will return a TEST END signal to the meter control interface logic which will become "ready."

The CONTROL and STATUS register READY bit is now asserted, indicating that the test is complete. The data word can now be read. The "error count" is the total number of errors since the last test in which RESET AT START was commanded.

If the 12-bit binary error counter in the meter control interface overflows at any time, the COUNT OVERFLOW bit is asserted. In addition, if the BERC has currently detected a loss of sync lock, the IN/OUT OF SYNC bit will indicate this loss. If the BERC had detected sync loss during the current test, but had detected that sync lock had been recovered before the end of this test, the SYNC RECOVERED bit will be set. A valid bit error test will be determined to have occurred at the end of the test if all of the following conditions are met:

1. OVERFLOW bit is off ("zero")
2. IN/OUT OF SYNC bit is on ("one")
3. SYNC RECOVERED bit is off ("zero").

This Page Intentionally Left Blank

SECTION IV

SOFTWARE DESCRIPTION

1. INTRODUCTION

Additional software has been added to the CSEL system to control the bit error rate counters (BERCs) when in remote mode. CSEL now supports 20 additional system parameters and contains two new procedures used to control BERC tests.

2. RUNNING A BERC TEST (REMOTE MODE)

To run a BERC test, the user enters the desired values into the BERC parameters. The contents of a BERC parameter may be examined by specifying the parameter's name followed by a question mark (?), as described for the other system parameters. Once the BERC parameters are entered, the CHECK directive will initiate a BERC test, provided that:

1. The BERC is enabled
2. There is no real-time run pending
3. Maximum time allowed, block size, and data rate have been entered
4. There is sufficient time for at least one block.

If both BERCs are enabled, the Model 1210A will be run first, and then the Model 3000. A BERC test will be terminated for the following reasons:

1. Minimum number of errors has been detected (normal termination)
2. Insufficient time for one more block
3. Lost synch
4. Error count overflow
5. BERC set to manual
6. Excessive block count (more than 30,000 blocks)

Note that the BERCs must be set to REMOTE for a test to be run, and that a run may be aborted by setting the BERC to MANUAL.

3. BERC PARAMETERS

Following are descriptions of the various BERC parameters. See Table 3 for a brief summary.

B1SW, B2SW - Enable a BERC test for 1210A and 3000, respectively.

B1DT, B1RS, B1SY - Set Data Terminal Ready, Request to Send, and Synchronous/Asynchronous switches on 1210A.

B1XM - Set transmitter clock and data levels on 1210A. Levels can be Polar (POL), Positive-Neutral (POS), or Negative-Neutral (NEG).

B1RC - Set receiver clock and data levels on 1210A (see B1XM).

B2DA - First parameter selects TTL or MIL 188 interface circuit for both transmitted and received data on 3000. Second parameter allows data signal to be normal or inverted.

B2XC - Same as B2DA, but for transmitter clock.

B2RC - Same as B2DA, but for receiver clock.

B1PN, B2PN - Select 63-, 511-, or 2047-bit maximal length pseudo-random pattern.

B1DR - Select data rate in kbps for 1210A. Data rate will be rounded to one of following values: .075, .110, .150, .300, .600, 1.2, 1.8, 2.4, 4.8, 7.2, or 9.6. "EXT" may be appended to signify an external data rate.

B2DR - Specify data rate in kbps for 3000.

B1SZ, B2SZ - Select block size in kilobits. Selected block size will be rounded to one of allowable values in Table 4.

TABLE 3

SUMMARY OF BERC PARAMETERS

Name	Values	Meaning
B1SW	ON/OFF	Enable BERC 1210A Test
B1DT	ON/OFF	Data Terminal Ready
B1RS	ON/OFF	Request To Send
B1SY	SYNC/ASYN	Select Async or Synchronous Operation
B1XM	POL./POS/NEG, POL/POS/NEG	Transmitter Clock, Data Levels
B1RC	POL/POS/NEG, POL/POS/NEG	Receiver Clock, Data Levels
B1PN	63/511/2047	Pattern Select
B1SZ	$0 <$ Decimal Fraction $\leq 2 \times 10^5$	Block Size (Kilobits)
B1DR	$0 \leq$ Decimal Fraction $\leq$ 500(EXT)	Data Rate (kbps)
B1ER	$0 \leq$ Integer $\leq 30,000$	Minimum Number of Errors
B1TM	Hours, Minutes	Both Integers, Equivalent in Minutes, Must Be $\leq 30,000$. Maximum Time Allowed for a 1210A Test.
B2SW	ON/OFF	Enable BERC 3000 Test
B2DA	MIL/TTL, NORM/INV	Data Interface, Sense
B2XC	MIL/TTL, NORM/INV	Transmit Clock Interface, Sense
B2RC	MIL/TTL, NORM/INV	Receive Clock Interface, Sense
B2PN	63/511/2047	Pattern Select
B2SZ	$0 <$ Decimal Fraction $\leq 2 \times 10^6$	Block Size (Kilobits)
B2DR	$0 \leq$ Decimal Fraction $\leq 10,000$	Data Rate (kbps)
B2ER	$0 \leq$ Integer $\leq 30,000$	Minimum Number of Errors
B2TM	Hours, Minutes	Both Integers, Equivalent in Minutes, Must Be $\leq 30,000$. Maximum Time Allowed for a 3000 Test.

TABLE 4

ALLOWABLE BLOCK SIZES

Code	Block Size
Model 1210A	
0	1 x Pattern Length
1	2 x Pattern Length
2	4 x Pattern Length
3	8 x Pattern Length
4	16 x Pattern Length
5	32 x Pattern Length
6	10^3 bits
7	10^4 bits
8	10^5 bits
9	10^6 bits
10	10^7 bits
11	10^8 bits
Model 3000	
0	10^2 bits
1	10^3 bits
2	10^4 bits
3	10^5 bits
4	10^6 bits
5	10^7 bits
6	10^8 bits
7	10^9 bits

B1ER, B2ER - Select minimum number of errors to be detected. Successive blocks will be processed until either minimum number of errors has been reached, there is insufficient time to run a block, or a fault has occurred. If zero errors are selected, only one block will be processed.

B1TM, B2TM - Select maximum time for a BERC test, in hours and minutes. Data rate, block size, and pattern length are used to determine amount of time required for one block. After each block is processed, a check is made to determine if there is enough time for one more block. If not, the test is terminated. This time-out check is disabled if maximum time is set to zero.

4. BERC FILE

After a BERC run is terminated for any reason other than set manual, the BERC number (1210A or 3000), number of blocks, kilobits, errors, and percentage of bits in error, are printed on the line printer. Also, an eight-word record is entered into the BERC file, with the format as shown in Figure 7.

The BERC file exists on the disk in either a Fade or Hop pattern file.

5. BERC PROCEDURES AND SUBROUTINES

The BERC tests are controlled by two procedures, BERC1 and BERC2, which call two subroutines, XMIT and BFILE. These procedures and subroutines are described on the following pages, together with flow charts and program listings.

Word #		
1	Number of Blocks (Integer)	
2	Number of kilobits (Single Precision Real)	
3		
4	Number of Errors (Integer)	
5	Data Rate (Single Precision Real)	
6	bps	
7	Status	BERC Number
8	Block Size	Pattern

BERC Number = 1 Model 1210A

= 2 Model 3000

Status = 0 Normal Termination

= 1 Insufficient Time

= 2 Lost Synch

= 3 Error Count Overflow

= 4 Excessive Block Count

Pattern = 3 63 bits

= 4 511 bits

= 5 2047 bits

Figure 7. BERC File Record

Procedure BERC1

Procedure BERC1 builds and sends control words to the 1210A, and runs a BERC test if B1SW is on.

Every time BERC1 is executed, it packs B1DT, B1RS, B1SY, B1XM, B1RC, and B1PN into Data Generator Control A and sends it to the UDC-11 via subroutine XMIT. If the data rate, B1DR, has been changed since the last execution of BERC1, it is sent to Data Generator Control B.

If B1SW is on, a BERC test is run. Successive blocks of pseudorandom signals are processed until either:

1. Minimum number of bit errors (B1ER) has occurred
2. Maximum time (B1TM) has been exceeded
3. Error in system operation occurs.

At the end of a run, the number of blocks, bits, errors, and percentage errors are printed out, and BFILE is called to store a record in the BERC file or disk.

The possible errors that may occur are:

UDC-11 errors (1ERR(1) = 1)

Read error (IERR(2) = 137)

Write error (IERR(2) = 237)

Verification error (IERR(2) = 337)

System error (IERR(1) = 3)

Wait error (IERR(2) = 100)

BERC error (IERR(1) = 4)

BERC1 in manual (IERR(2) = 137)

BERC1 not ready (IERR(2) = 237)

BERC1 sync recovered, will not reset (IERR(2) = 337)

Directory search error (IERR(2) = 400)

No room for BERC file (IERR(2) = 500)

BERC file full (IERR(2) = 600)

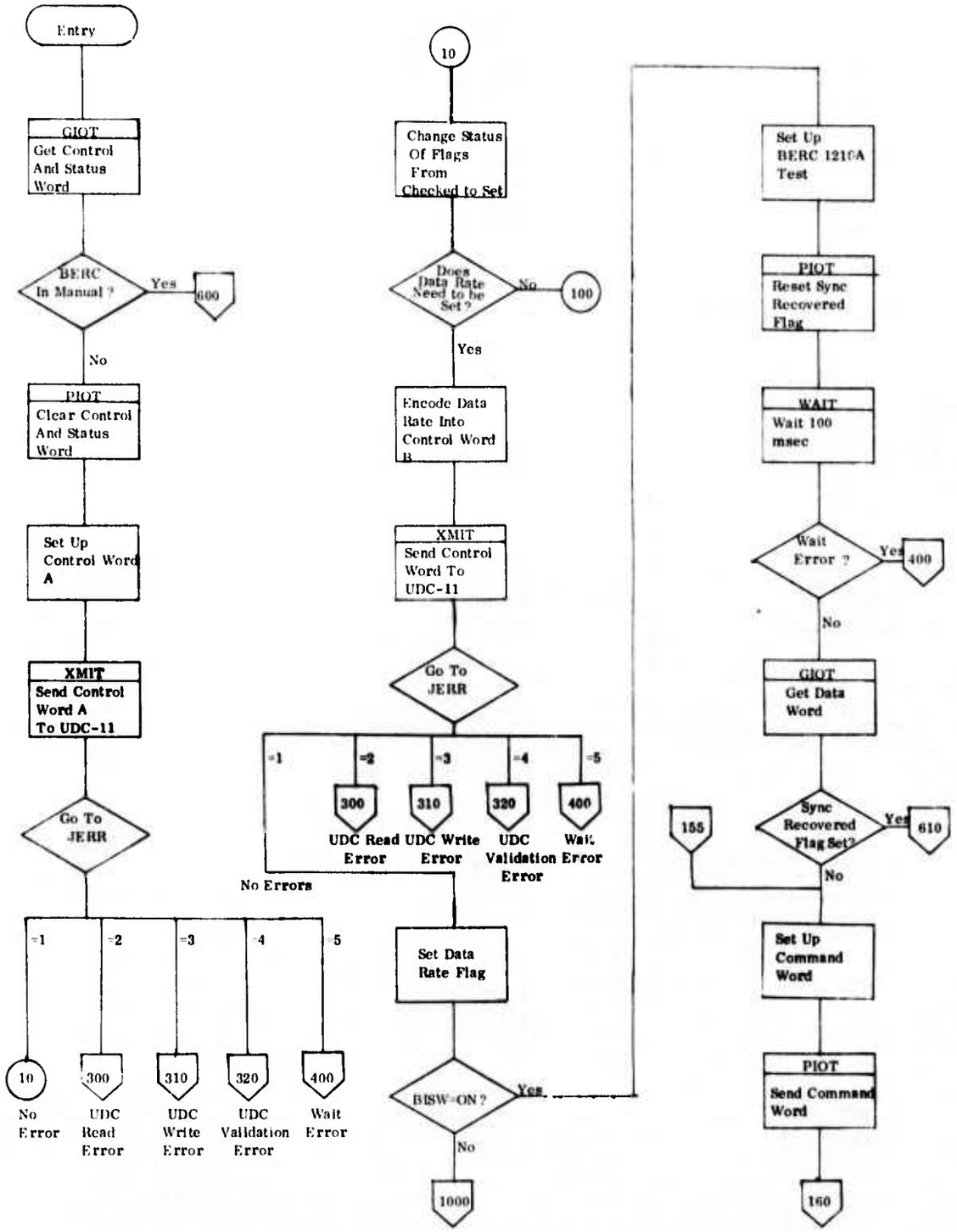
Run terminated due to lost sync

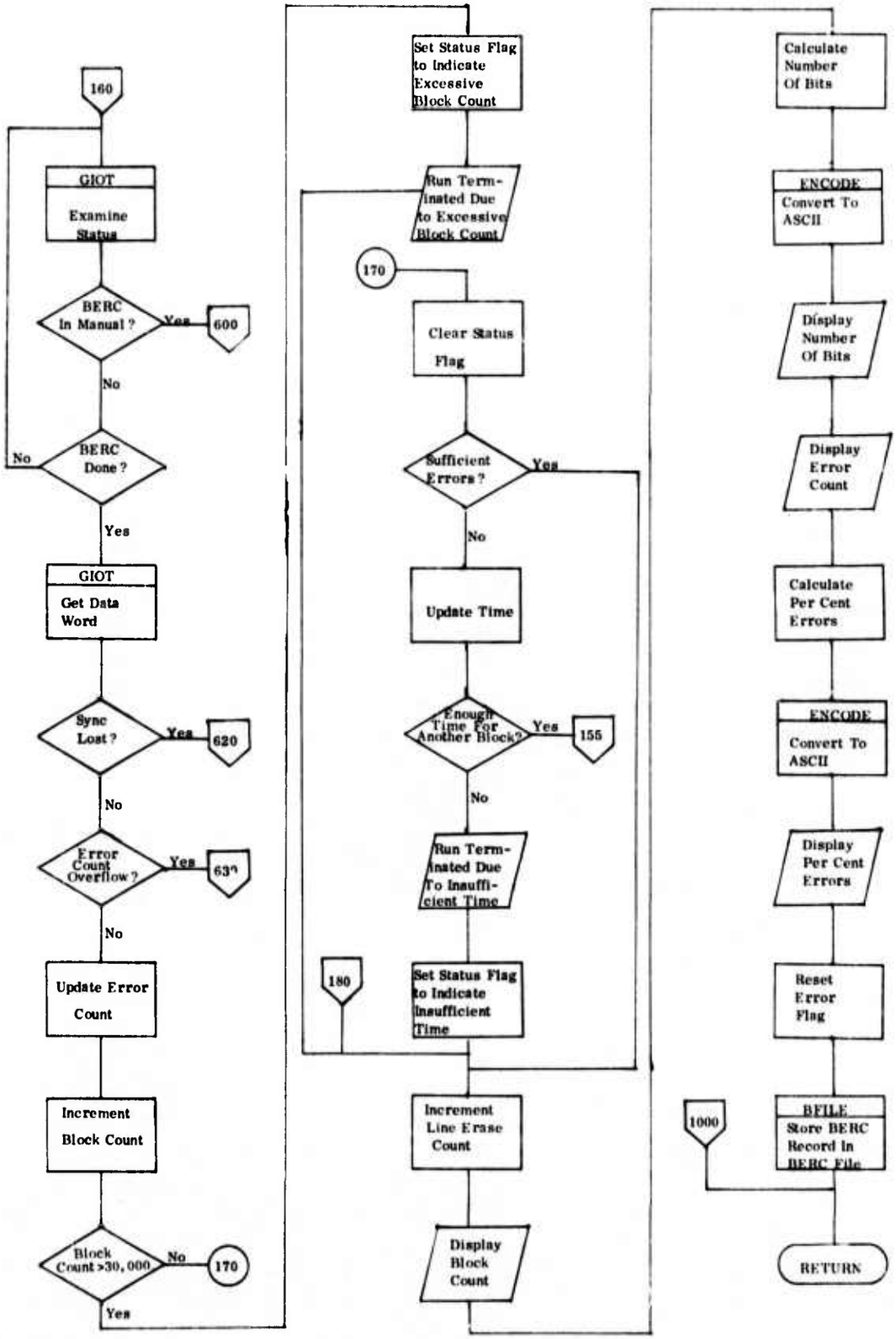
Run terminated due to insufficient time

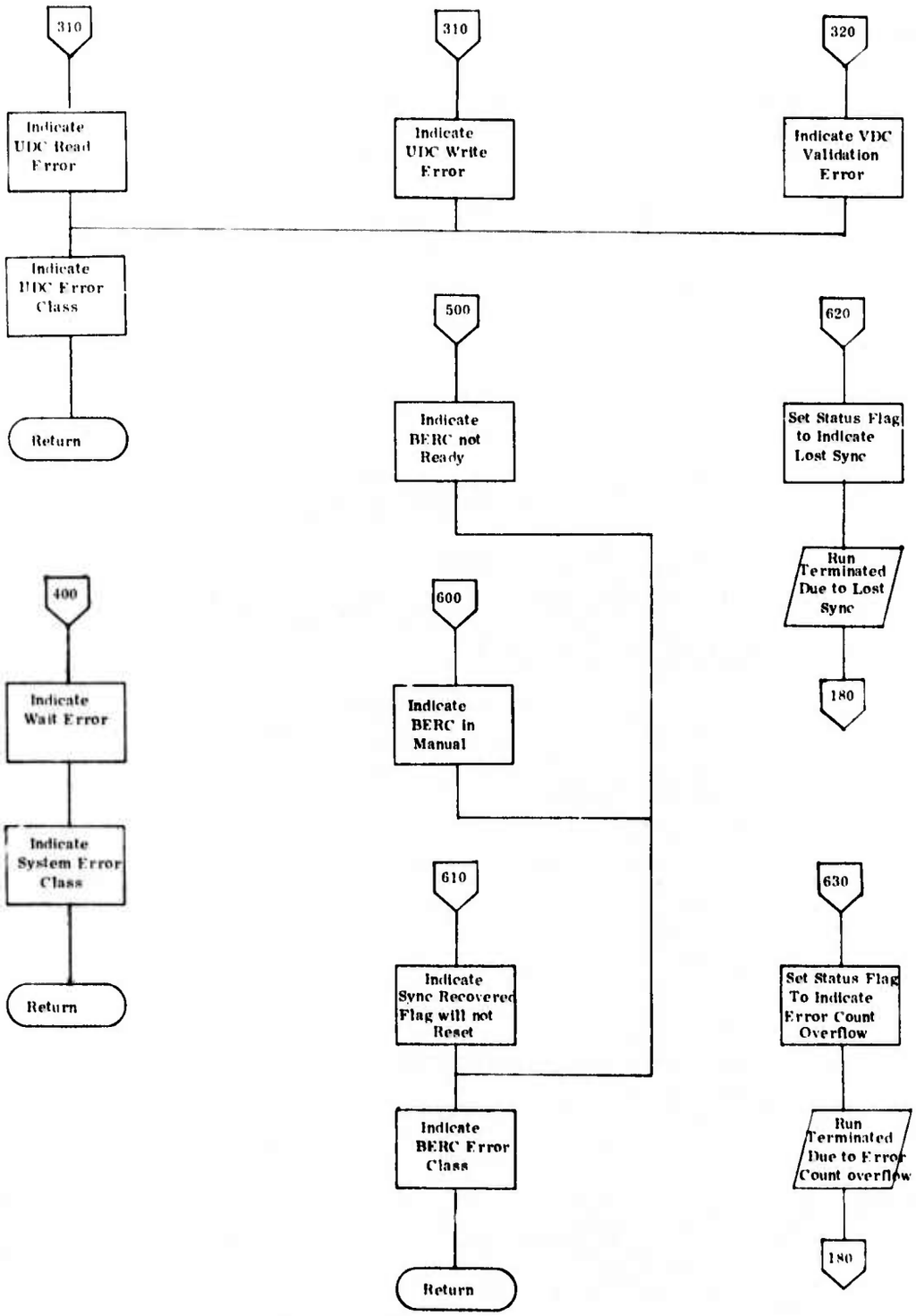
Run terminated due to error count overflow

Run terminated due to excessive block count

The last four error messages are printed by BERC1 and do not return an error code.


Entry
GIOT
Get Control And Status Word
BERC In Manual ?
Yes
600
No
PIOT
Clear Control And Status Word
Set Up Control Word A
XMIT
Send Control Word A To UDC-11
Go To JERR
=1
=2
=3
=4
=5
10
No Error
300
UDC Read Error
310
UDC Write Error
320
UDC Validation Error
400
Wait Error
10
Change Status Of Flags From Checked to Set
Does Data Rate Need to be Set ?
No
100
Yes
Encode Data Rate Into Control Word B
XMIT
Send Control Word To UDC-11
Go To JERR
=1
=2
=3
=4
=5
300
UDC Read Error
310
UDC Write Error
320
UDC Validation Error
400
Wait Error
No Errors
Set Data Rate Flag
BISW=ON ?
Yes
No
1000
Set Up BERC 1210A Test
PIOT
Reset Sync Recovered Flag
WAIT
Wait 100 msec
Wait Error ?
Yes
400
No
GIOT
Get Data Word
155
Sync Recovered Flag Set?
Yes
610
No
Set Up Command Word
PIOT
Send Command Word
160


160
GIOT
Examine Status
BERC In Manual ?
Yes
600
No
BERC Done ?
No
Yes
GIOT
Get Data Word
Sync Lost ?
Yes
620
No
Error Count Overflow ?
Yes
630
No
Update Error Count
Increment Block Count
Block Count >30, 000
No
170
Yes
Set Status Flag to Indicate Excessive Block Count
Run Terminated Due to Excessive Block Count
170
Clear Status Flag
Sufficient Errors ?
Yes
No
Update Time
Enough Time For Another Block?
Yes
155
No
Run Terminated Due To Insufficient Time
180
Set Status Flag to Indicate Insufficient Time
Increment Line Erase Count
Display Block Count
Calculate Number Of Bits
ENCODE
Convert To ASCII
Display Number Of Bits
Display Error Count
Calculate Per Cent Errors
ENCODE
Convert To ASCII
Display Per Cent Errors
Reset Error Flag
1000
BFILE
Store BERC Record In BERC File
RETURN


310
Indicate UDC Read Error
Indicate UDC Error Class
Return
310
Indicate UDC Write Error
320
Indicate VDC Validation Error
500
Indicate BERC not Ready
600
Indicate BERC in Manual
610
Indicate Sync Recovered Flag will not Reset
Indicate BERC Error Class
Return
620
Set Status Flag to Indicate Lost Sync
Run Terminated Due to Lost Sync
180
630
Set Status Flag To Indicate Error Count Overflow
Run Terminated Due to Error Count overflow
180
400
Indicate Wait Error
Indicate System Error Class
Return

```
FORTRAN V004A                    08:07:11    04-APR-75    PAGE    1

      C      **PROCEDURE FOR 1210A BERC**
      C THIS PROCEDURE ALWAYS SETS UP CONTROL WORD A.
      C CONTROL WORD B IS SET UP IF REQUIRED.  IF SIBW
      C IS ON, THE BERC IS RUN UNTIL EITHER THERE IS AN
      C ERROR, THERE IS INSUFFICIENT TIME, OR THE MINIMUM ERRORS,
      C THE NUMBER OF BLOCKS, BITS, ERRORS, AND %ERRORS ARE DISPLAYED.
            COMMON/IMVTAB/ISPAC(174),IFLG,BERC,POLAR,SENSE,PATN,
           1SIZE1,SIZE2,MAXTIM,MINERR,DATRAT
            REAL*8 REAVAL
             REAL*4 DATRAT(2)
             INTEGER MAXTIM(2),MINERR(2)
             BYTE BERC(5),POLAR(2,2),SENSE(3),PATN(2),SIZE1,SIZE2
             BYTE IFLG(80)
            BYTE PVAL(12)
            COMMON/ARGS/ISKIP(20),IERR(2)
            COMMON/COMFLG/ICOND,IEAS,JERR,NEWFIL
            REAL*4 BSVAL(12)
            REAL*4 LIMIT(11)
            INTEGER DGCA,DGCB,DGI,CSW,CMND,DATWD
            INTEGER PATVAL(3),IARRAY(6),CKD,SET,ON,OFF
            DIMENSION KEY(11)
            DATA IARRAY/63,64,65,66,67,71/
            DATA PATVAL/63,511,2047/
            DATA BSVAL/1.,2.,4.,8.,16.,32.,1E3,1E4,1E5,1E6,1E7,1E8/
            DATA LIMIT/8000.,5600.,3200.,2000.,1400.,800.,
           *400.,200.,130.,90.,0./
            DATA KEY/9,8,7,6,5,4,3,2,1,0,11/
            DATA DGCA/"70/DGCB/"71/DGI/"22/
            DATA CSW/"167730/CMND/"167732/DATWD/"167734/
            DATA MSKMAN/"100000/MSKRDY/"200/MSKSYN/"40000/MSKOVF/"10000/
            DATA IZERO/"0/IRESET/"200/ITEST/"560/
            DATA CKD/8/SET/4/ON/1/OFF/0/
            DATA IA/1/IB/2/
            DATA IDIG1,IDIG2,IDIG3/3*0/
            CALL GIOY(CSW,JDIN)
            IF(IAND(JDIN,MSKMAN).NE.0) GO TO 600
            CALL PIOY(CSW,IZERO)
      C       SET UP CONTROL WORD A
            JDOUT=0
            ITMP=BERC(3)
            CALL PACK(1,14,ITMP,JDOUT)
            ITMP=BERC(4)
            CALL PACK(1,13,ITMP,JDOUT)
            ITMP=BERC(5)
            CALL PACK(1,12,ITMP,JDOUT)
            ITMP=POLAR(1,1)
            CALL PACK(2,10,ITMP,JDOUT)
            ITMP=POLAR(2,1)
            CALL PACK(2,8,ITMP,JDOUT)
            ITMP=POLAR(1,2)
            CALL PACK(2,6,ITMP,JDOUT)
            ITMP=POLAR(2,2)
            CALL PACK(2,4,ITMP,JDOUT)
            ITMP=PATN(1)
            CALL PACK(3,0,ITMP,JDOUT)
```

```
C       SEND CONTROL WORD A TO UDC
        CALL XMIT(DGCA,DGI,JDOUT,IA,JERR)
        GO TO (10,300,310,320,400) JERR
   10   DO 20 I=1,6
        IF(IFLG(IARRAY(I)).EQ.CKD) IFLG(IARRAY(I))=SET
   20   CONTINUE
        IF(IFLG(79).EQ.SET) GO TO 100
C         ENCODE DATA RATE
        JDOUT=10
        DO 30 I=1,11
        IF(DATRAT(1).LT.LIMIT(I)) JDOUT=KEY(I)
   30   CONTINUE
C       SEND CONTROL WORD B TO UDC
        CALL XMIT(DGCB,DGI,JDOUT,IB,JERR)
        GO TO (40,300,310,320,400) JERR
   40   IFLG(79)=SET
  100   CONTINUE
        IF(BERC(1).EQ.OFF) GO TO 1000
C         SET UP BERC 1210A TEST
        BLOCK=BSVAL(SIZE1+1)
        IF(SIZE1.GE.6) GO TO 110
        BLOCK=BLOCK*PATVAL(PATN(1)-2)
  110   TINCR=BLOCK/ABS(DATRAT(1))/60
        TIME=TINCR
        NBLK=0
        ICOUNT=0
C         RESET SYNC RECOVERED FLAG
  150   CALL PIOT(CMND,IRESET)
        CALL WAIT(100,1,JWFLG)
        IF(JWFLG.NE.1) GO TO 400
        CALL GIOT(DATWD,JDIN)
        IF(IAND(JDIN,MSKSYN).NE.0) GO TO 610
C         SEND COMMAND WORD--START TEST
  155   CONTINUE
        JDOUT=ITEST
        ITMP=SIZE1
        CALL PACK(4,0,ITMP,JDOUT)
        CALL PIOT(CMND,JDOUT)
  160   CONTINUE
        CALL GIOT(CSW,JDIN)
        IF(IAND(JDIN,MSKMAN).NE.0) GO TO 600
        IF(IAND(JDIN,MSKRDY).EQ.0) GO TO 160
        CALL GIOT(DATWD,JDIN)
        IF(IAND(JDIN,MSKSYN).NE.0) GO TO 620
        IF(IAND(JDIN,MSKOVF).NE.0) GO TO 630
        ICOUNT=ICOUNT+IAND(JDIN,"7777)
        NBLK=NBLK+1
        IF(NBLK.NE.30000) GO TO 170
        ISTAT=4
        WRITE(2,899)
  899   FORMAT(" RUN TERMINATED DUE TO EXCESSIVE BLOCK COUNT"/)
        GO TO 180
  170   CONTINUE
        ISTAT=0
        IF(ICOUNT.GE.MINERR(I)) GO TO 200
C         INSUFFICIENT ERRORS, TRY ANOTHER BLOCK
```

```
      TIME=TIME+TINCR
      IF(TIME.LE.MAXTIM(1).OR.MAXTIM(1).EQ.0) GO TO 155
C       INSUFFICIENT TIME FOR ANOTHER BLOCK
      WRITE (2,900)
  900 FORMAT(' RUN TERMINATED DUE TO INSUFFICIENT TIME'/)
      ISTAT=1
  180 IEAS=IEAS+2
C       DISPLAY # BLOCKS, BITS, ERRORS, %ERRORS
  200 CONTINUE
      WRITE(5,210) NBLK
  210 FORMAT('0',8X,'BERC  1210A'/1X,I12,'  BLOCKS')
      BITS=BLOCK*NBLK/1000.0
      REAVAL=BITS
      CALL ENCODE(REAVAL,PVAL,0)
      WRITE(5,220) PVAL
  220 FORMAT(1X,12A1,'  KBITS')
      WRITE(5,230) ICOUNT
  230 FORMAT(1X,I12,'  ERRORS')
      IF(BITS.NE.0) GO TO 250
      REAVAL=0
      GO TO 260
  250 REAVAL=ICOUNT*100.0/(BITS*1000)
  260 CALL ENCODE(REAVAL,PVAL,0)
      WRITE(5,240) PVAL
  240 FORMAT(1X,12A1,'% ERRORS'/)
      RATE=ABS(DATRAT(1))
      IERR(1)=0
      CALL BFILE(NBLK,BITS,ICOUNT,RATE,1,ISTAT,PATN(1),SIZE1)
 1000 CONTINUE
      CALL RETURN
C       UDC ERROR
  300 CONTINUE
      IERR(2)=137
      GO TO 350
  310 CONTINUE
      IERR(2)=237
      GO TO 350
  320 CONTINUE
      IERR(2)=337
  350 IERR(1)=1
      CALL RETURN
C       WAIT ERROR
  400 CONTINUE
      IERR(1)=3
      IERR(2)=100
      CALL RETURN
C       BERC ERRORS
  500 CONTINUE
      IERR(2)=237
  550 IERR(1)=4
      CALL RETURN
  600 CONTINUE
      IERR(2)=137
      GO TO 550
  610 CONTINUE
      IERR(2)=337
```

```
FORTRAN V004A                    08:07:11    04-APR-75   PAGE    4

          GO TO 550
      620 CONTINUE
          ISTAT=2
          WRITE (2,910)
      910 FORMAT(' RUN TERMINATED DUE TO LOST SYNC'/)
          GO TO 180
      630 CONTINUE
          ISTAT=3
          WRITE (2,920)
      920 FORMAT(' RUN TERMINATED DUE TO ERROR COUNT OVERFLOW'/)
          GO TO 180
          END

     ROUTINES CALLED:
     GIOT  , IAND  , PIOT  , PACK  , XMIT  , ABS   , WAIT
     ENCODE, BFILE , RETURN

     SWITCHES = /ON,/SU,/CO

     BLOCK        LENGTH
     MAIN,   1353    (005222)*
     IMVTAB  230     (000714)
     ARGS    22      (000054)
     COMFLG  4       (000010)

     **COMPILER ----- CORE**
        PHASE        USED  FREE
     DECLARATIVES  01086 16788
     EXECUTABLES   02008 15858
     ASSEMBLY      02054 18729
```

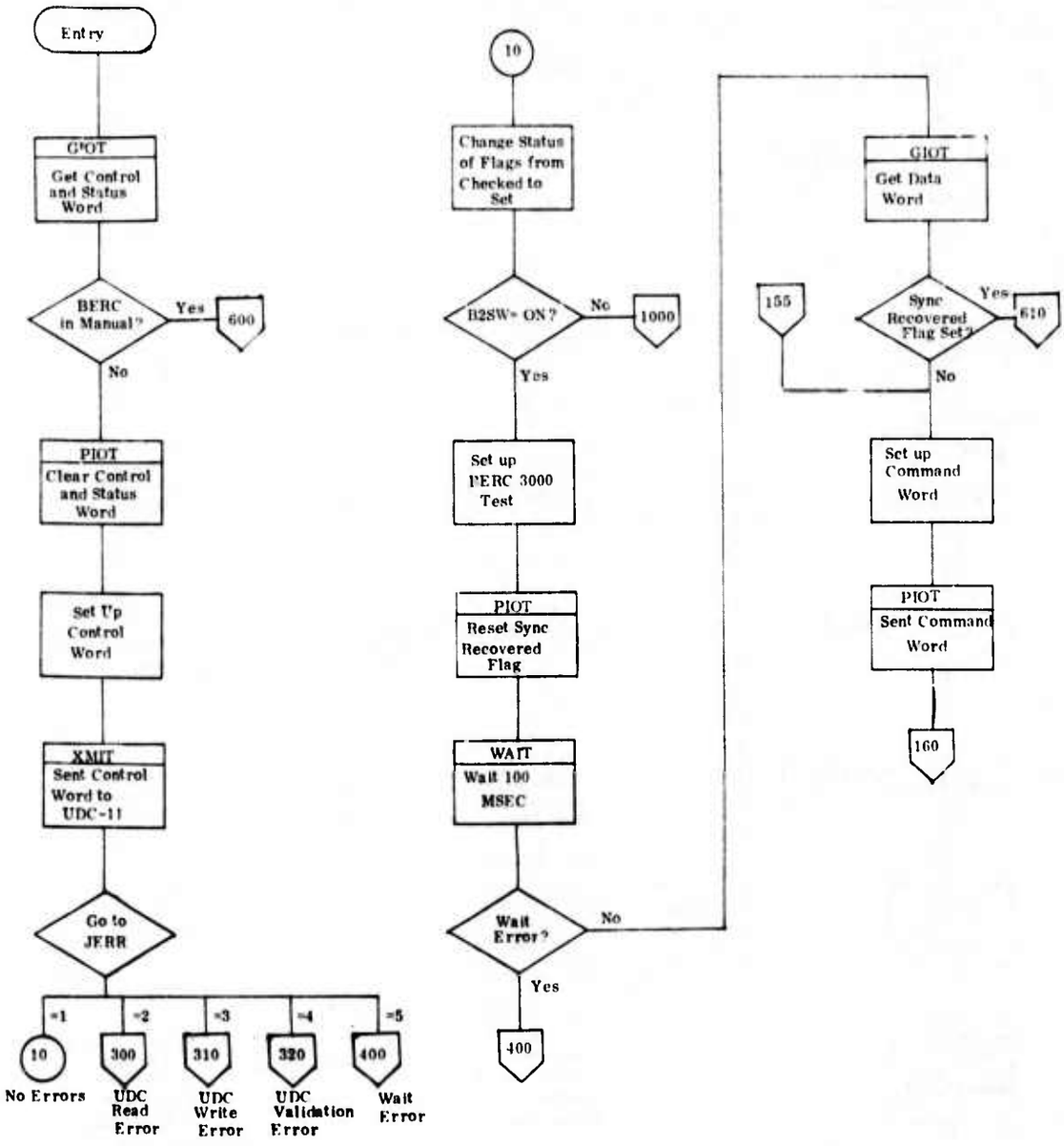
Procedure BERC2

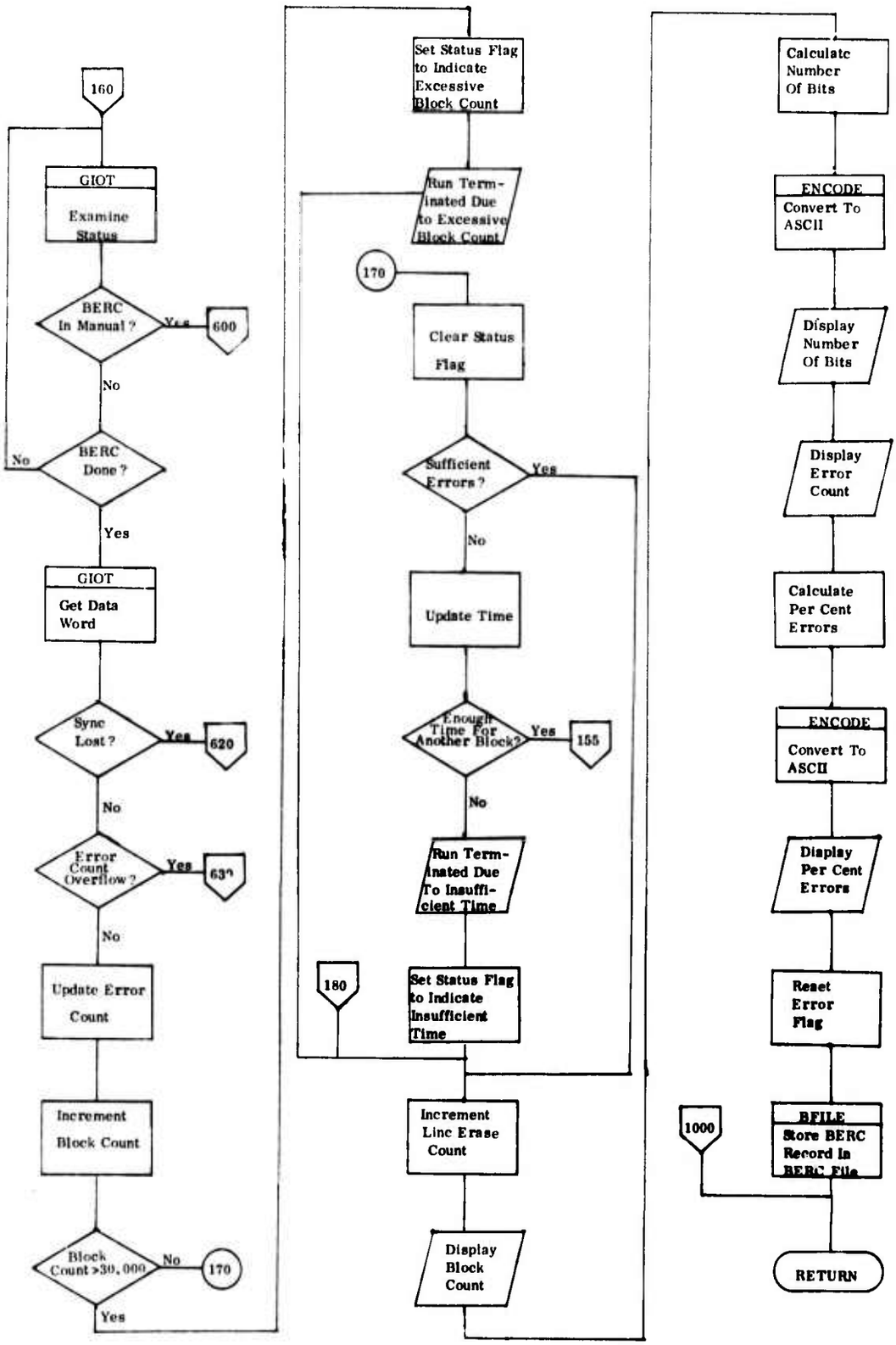
Procedure BERC2 builds and sends control words to the BERC 3000, and runs a BERC test if B2SW is on.

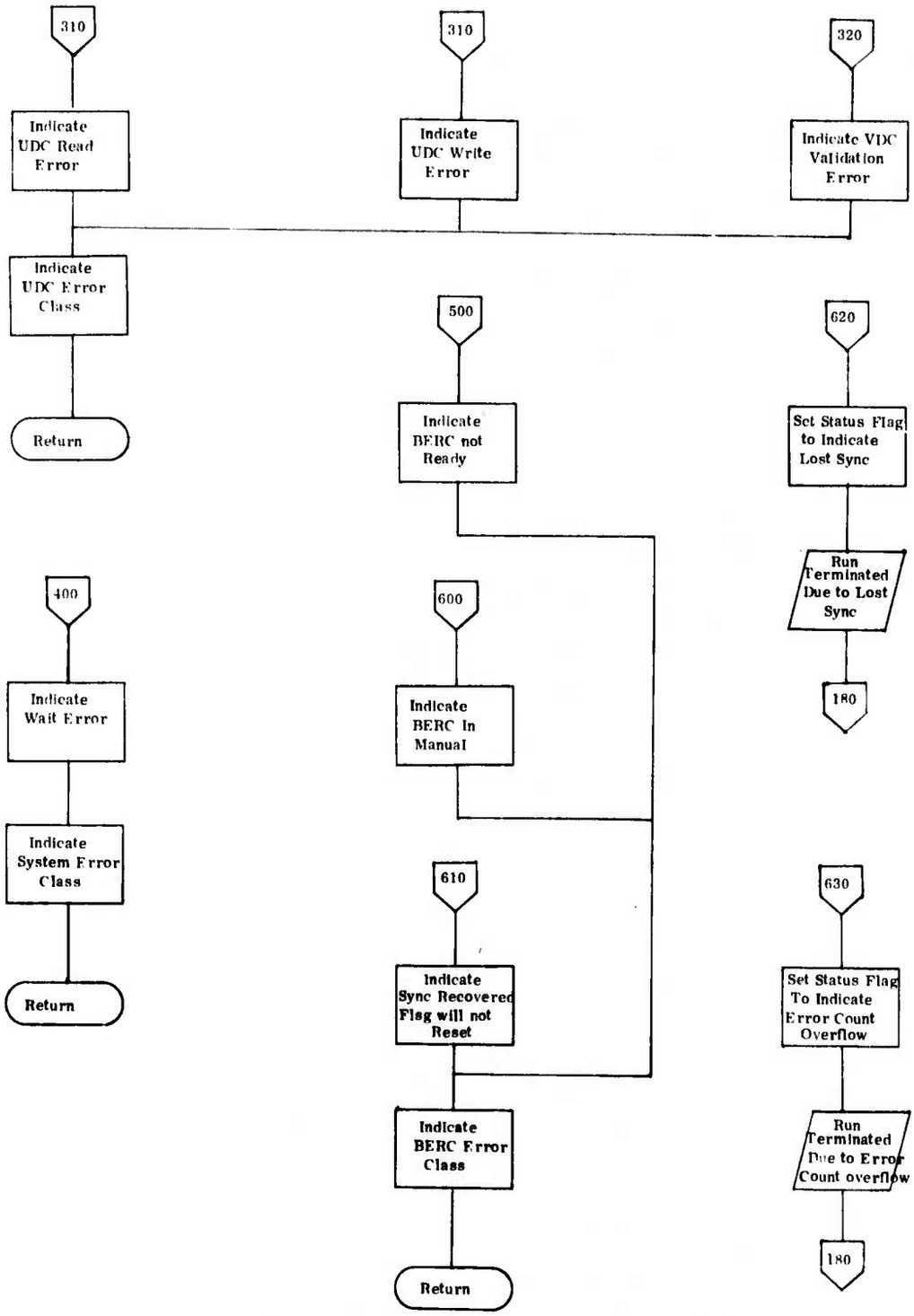
BERC2 is similar to BERC1. The main difference is that it packs B2DA, B2XC, B2RC, and B2PN into the single Data Generator Control Word for the BERC 3000.

B2ER is used for the minimum errors and B2TM is used for the maximum time. B2DR is only used to determine the length of time for one block.

Also, in BERC2, IERR(2) will be set to 138, 238, and 338 instead of 137, 237, and 337 as in BERC1.


Entry
GPOT
Get Control and Status Word
BERC in Manual?
Yes
600
No
PIOT
Clear Control and Status Word
Set Up Control Word
XMIT
Sent Control Word to UDC-11
Go to JERR
=1
=2
=3
=4
=5
10
300
310
320
400
No Errors
UDC Read Error
UDC Write Error
UDC Validation Error
Wait Error
10
Change Status of Flags from Checked to Set
B28W= ON?
No
1000
Yes
Set up PERC 3000 Test
PIOT
Reset Sync Recovered Flag
WAIT
Wait 100 MSEC
Wait Error?
No
Yes
400
155
GIOT
Get Data Word
Sync Recovered Flag Set?
Yes
610
No
Set up Command Word
PIOT
Sent Command Word
160


160
GIOT
Examine Status
BERC In Manual ?
Yes
600
No
BERC Done ?
No
Yes
GIOT
Get Data Word
Sync Lost ?
Yes
620
No
Error Count Overflow ?
Yes
630
No
Update Error Count
Increment Block Count
Block Count >30,000
No
170
Yes
Set Status Flag to Indicate Excessive Block Count
Run Terminated Due to Excessive Block Count
170
Clear Status Flag
Sufficient Errors ?
Yes
No
Update Time
Enough Time For Another Block?
Yes
155
No
Run Terminated Due To Insufficient Time
180
Set Status Flag to Indicate Insufficient Time
Increment Line Erase Count
Display Block Count
Calculate Number Of Bits
ENCODE
Convert To ASCII
Display Number Of Bits
Display Error Count
Calculate Per Cent Errors
ENCODE
Convert To ASCII
Display Per Cent Errors
Reset Error Flag
1000
BFILE
Store BERC Record in BERC File
RETURN


310
Indicate UDC Read Error
310
Indicate UDC Write Error
320
Indicate VDC Validation Error
Indicate UDC Error Class
Return
500
Indicate BERC not Ready
620
Set Status Flag to Indicate Lost Sync
Run Terminated Due to Lost Sync
180
400
Indicate Wait Error
Indicate System Error Class
Return
600
Indicate BERC In Manual
610
Indicate Sync Recovered Flag will not Reset
Indicate BERC Error Class
Return
630
Set Status Flag To Indicate Error Count Overflow
Run Terminated Due to Error Count overflow
180

```
FORTRAN V004A                    00:00:46    04-APR-75    PAGE    1

      C     **PROCEDURE FOR 3000 BERC**
      C THIS PROCEDURE ALWAYS SETS UP THE CONTROL WORD.  IF B2SW IS ON,
      C THE BERC IS RUN UNTIL THERE IS EITHER AN ERROR, INSUFFICIENT TIME,
      C OR THE MINIMUM ERRORS, THE NUMBER OF BITS, BLOCKS, ERRORS, AND
      C %ERRORS ARE DISPLAYED.
            COMMON/IMVTAB/ISPAC(174),IFLG,BERC,POLAR,SENSE,PATN,
           1SIZE1,SIZE2,MAXTIM,MINERR,DATRAT
            REAL*8 REAVAL
             REAL*4 DATRAT(2)
             INTEGER MAXTIM(2),MINERR(2)
             BYTE BERC(5),POLAR(2,2),SENSE(3),PATN(2),SIZE1,SIZE2
             BYTE IFLG(80)
            BYTE PVAL(12)
            COMMON/ARGS/ISKIP(20),IERR(2)
            COMMON/COMFLG/ICOND,IEAS,JERR,NEWFIL
            REAL*4 BSVAL(8)
            INTEGER DGC,DGI,CSW,CMND,DATWD
            INTEGER IARRAY(4),CKD,SET,ON,OFF
            EQUIVALENCE(ISIZE,SIZE1)
            EQUIVALENCE(IPAT,PATN)
            DATA BSVAL/1E2,1E3,1E4,1E5,1E6,1E7,1E8,1E9/
            DATA IARRAY/68,69,70,72/
            DATA DGC/"73/DGI/"23/CSW/"167720/CMND/"167722/DATWD/"167724/
            DATA MSKMAN/"100000/MSKRDY/"200/MSKSYN/"40000/MSKOVF/"10000/
            DATA IZERO/"0/IRESET/"200/ITEST/"560/
            DATA CKD/8/SET/4/ON/1/OFF/0/
            DATA IDIG1,IDIG2,IDIG3/3*0/

            CALL GIOT(CSW,JDIN)
            IF(IAND(JDIN,MSKMAN).NE.0) GO TO 600
            CALL PIOT(CSW,IZERO)
      C       SET UP CONTROL WORD
            JDOUT=0
            ITMP=SENSE(1)
            CALL PACK(2,8,ITMP,JDOUT)
            ITMP=SENSE(2)
            CALL PACK(2,6,ITMP,JDOUT)
            ITMP=SENSE(3)
            CALL PACK(2,4,ITMP,JDOUT)
            ITMP=PATN(2)
            CALL PACK(3,0,ITMP,JDOUT)
      C       SEND CONTROL WORD TO UDC
            CALL XMIT(DGC,DGI,JDOUT,0,JERR)
            GO TO (10,300,310,320,400) JERR
         10 DO 20 I=1,4
            IF(IFLG(IARRAY(I)).EQ.CKD) IFLG(IARRAY(I))=SET
         20 CONTINUE
            IF(BERC(2).EQ.OFF) GO TO 1000
      C       SET UP BERC 3002 TEST
            ISIZ2=0
            CALL UNPACK(8,8,ISIZE,ISIZ2)
            BLOCK=BSVAL(ISIZ2+1)
        110 TINCR=BLOCK/ABS(DATRAT(2))/60
            TIME=TINCR
            NBLK=0
```

```
      ICOUNT=0
C       RESET SYNC RECOVERED FLAG
  150 CALL PIOT(CMND,IRESET)
      CALL WAIT(100,1,JWFLG)
      IF(JWFLG.NE.1) GO TO 400
      CALL GIOT(DATWD,JDIN)
      IF(IAND(JDIN,MSKSYN).NE.0) GO TO 610
C       SEND COMMAND WORD--START TEST
  155 CONTINUE
      JDOUT=ITEST
      CALL PACK(4,0,IBIZ2,JDOUT)
      CALL PIOT(CMND,JDOUT)
  160 CONTINUE
      CALL GIOT(CSW,JDIN)
      IF(IAND(JDIN,MSKMAN).NE.0) GO TO 600
      IF(IAND(JDIN,MSKRDY).EQ.0) GO TO 160
      CALL GIOT(DATWD,JDIN)
      IF(IAND(JDIN,MSKSYN).NE.0) GO TO 620
      IF(IAND(JDIN,MSKOVF).NE.0) GO TO 630
      ICOUNT=ICOUNT+IAND(JDIN,"7777)
      NBLK=NBLK+1
      IF(NBLK.NE.30000) GO TO 170
      ISTAT=4
      WRITE(2,899)
  899 FORMAT(' RUN TERMINATED DUE TO EXCESSIVE BLOCK COUNT'/)
      GO TO 180
  170 CONTINUE
      ISTAT=0
      IF(ICOUNT.GE.MINERR(2)) GO TO 200
C       INSUFFICIENT ERRORS, TRY ANOTHER BLOCK
      TIME=TIME+TINCR
      IF(TIME.LE.MAXTIM(2).OR.MAXTIM(2).EQ.0) GO TO 155
C       INSUFFICIENT TIME FOR ANOTHER BLOCK
      WRITE (2,900)
  900 FORMAT(' RUN TERMINATED DUE TO INSUFFICIENT TIME'/)
      ISTAT=1
  180 IEAS=IEAS+2
C       DISPLAY # BLOCKS, BITS, ERRORS, %ERRORS
  200 CONTINUE
      WRITE(5,210) NBLK
  210 FORMAT('0',8X,'BERC  3000'/1X,I12,'  BLOCKS')
      BITS=BLOCK*NBLK/1000.0
      REAVAL=BITS
      CALL ENCODE(REAVAL,PVAL,0)
      WRITE(5,220) PVAL
  220 FORMAT(1X,12A1,'  KBITS')
      WRITE(5,230) ICOUNT
  230 FORMAT(1X,I12,'  ERRORS')
      IF(BITS.NE.0) GO TO 250
      REAVAL=0
      GO TO 260
  250 REAVAL=ICOUNT*100.0/(BITS*1000)
  260 CALL ENCODE(REAVAL,PVAL,0)
      WRITE(5,240) PVAL
  240 FORMAT(1X,12A1,'% ERRORS'/)
      IPAT2=0
      IERR(1)=0
```

```
FORTRAN V004A                    00100146    04-APR-75   PAGE    3

      CALL UNPACK(8,8,IPAT,IPAT2)
      CALL BFILE(NBLK,BITS,ICOUNT,DATRAT(2),2,ISTAT,IPAT2,ISIZ2)
 1000 CONTINUE
      CALL RETURN
C       UDC ERROR
  300 CONTINUE
      IERR(2)=130
      GO TO 350
  310 CONTINUE
      IERR(2)=230
      GO TO 350
  320 CONTINUE
      IERR(2)=330
  350 IERR(1)=1
      CALL RETURN
C       WAIT ERROR
  400 CONTINUE
      IERR(1)=3
      IERR(2)=100
      CALL RETURN
C       BERC ERRORS
  500 CONTINUE
      IERR(2)=230
  550 IERR(1)=4
      CALL RETURN
  600 CONTINUE
      IERR(2)=130
      GO TO 550
  610 CONTINUE
      IERR(2)=330
      GO TO 550
  620 CONTINUE
      ISTAT=2
      WRITE (2,910)
  910 FORMAT(' RUN TERMINATED DUE TO LOST SYNC'/)
      GO TO 100
  630 CONTINUE
      ISTAT=3
      WRITE (2,920)
  920 FORMAT(' RUN TERMINATED DUE TO ERROR COUNT OVERFLOW'/)
      GO TO 100
      END

ROUTINES CALLED:
GINT  , IAND  , PIOT  , PACK  , XMIT  , UNPACK, ABS
WAIT  , ENCODF, BFILE , RETURN

SWITCHES = /ON,/8U,/CO

BLOCK        LENGTH
MAIN.   1111    (004256)*
IMVTAB  230     (000714)
ARGS    22      (000054)
COMFLG  4       (000010)

**COMPILER ----- CORE**
   PHASE        USED  FREE
DECLARATIVES 01006 16700
EXECUTABLES  01040 15910
ASSEMBLY     01930 10853
```

Subroutine BFILE

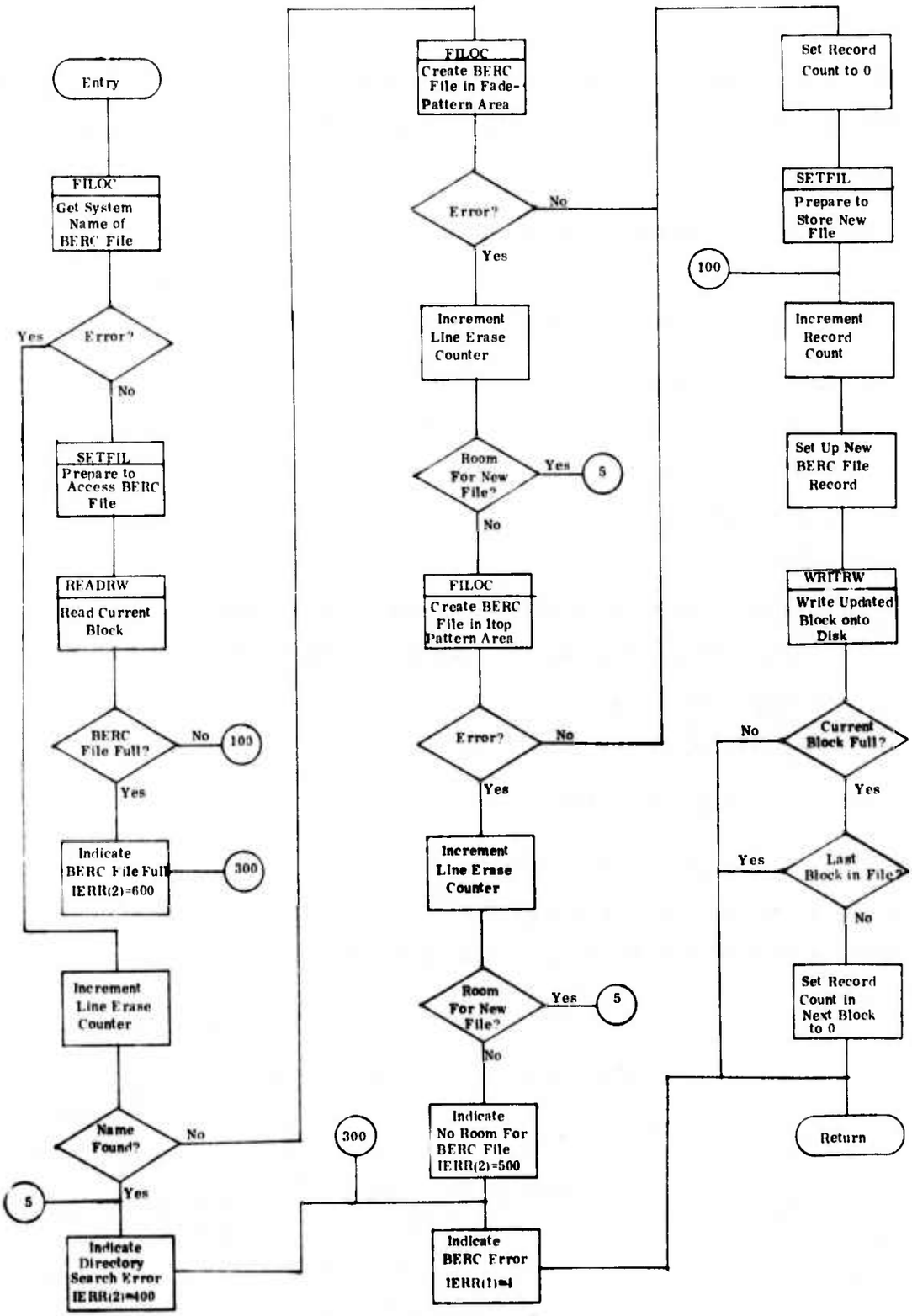
BFILE is called by BERC1 to enter a reeord of BERC data into the BERC File. If a BERC File does not exist, BFILE attempts to ereate it in either the Fade or Hop area on the Disk.

A BERC File consists of eight blocks of 256 words. Each block is organized into 32 records, the last of whieh contains only the number of records in that bloek. A BERC File record has eight words as follows:

1. Number of bloeks (Integer)
2., 3. Number of kilobits (single preeision)
4. Number of errors (Integer)
5., 6. Data Rate (bits per second)
7. Low Byte: BERC Number (1 or 2)
7. High Byte: Status (0 = No errors, 1 = Insuffieient time, 2 = Lost Sync, 3 = Error Count Overflow, 4 = Exeessive Bloek Count)
8. Low Byte: Pattern (Code)
8. High Byte: Bloek Size (Code)

Error codes are returned using IERR:

IERR(1) = 4 If there are any errors
IERR(2) = 400 If there is a directory search error
IERR(2) = 500 If there is no room for a BERC File
IERR(2) = 600 If BERC File is full.


Entry
FILOC
Get System Name of BERC File
Error?
Yes
No
SETFIL
Prepare to Access BERC File
READRW
Read Current Block
BERC File Full?
No
100
Yes
Indicate BERC File Full IERR(2)=600
300
Increment Line Erase Counter
Name Found?
No
Yes
5
Indicate Directory Search Error IERR(2)=400
300
FILOC
Create BERC File in Fade-Pattern Area
Error?
No
Yes
Increment Line Erase Counter
Room For New File?
Yes
5
No
FILOC
Create BERC File in Itop Pattern Area
Error?
No
Yes
Increment Line Erase Counter
Room For New File?
Yes
5
No
Indicate No Room For BERC File IERR(2)=500
Indicate BERC Error IERR(1)=4
Set Record Count to 0
SETFIL
Prepare to Store New File
100
Increment Record Count
Set Up New BERC File Record
WRITRW
Write Updated Block onto Disk
Current Block Full?
No
Yes
Last Block in File?
Yes
No
Set Record Count in Next Block to 0
Return

```
FORTRAN V004A                    08:10:09    04-APR-75    PAGE

      SUBROUTINE BFILE(NBLK,BITS,ICOUNT,RATE,NBERC,ISTAT,PATN,SIZE)
C       ENTER RECORD OF BERC DATA INTO BERC FILE
C         IERR(1)=4 IF ANY ERRORS
C         IERR(2)=400 IF DIRECTORY SEARCH ERROR
C         IERR(2)=500 IF NO ROOM FOR BERC FILE
C         IERR(2)=600 IF BERC FILE FULL
C
C       EACH BERC FILE RECORD HAS 8 WORDS:
C       1:    NUMBER OF BLOCKS (INTEGER)
C       2,3:  NUMBER OF KBITS (SINGLE PRECISION)
C       4:    NUMBER OF ERRORS (INTEGER)
C       5,6:  DATA RATE (SINGLE PRECISION)
C       7:    LOW BYTE: BERC NUMBER(1 OR 2);
C       7:    HIGH BYTE: STATUS (0=NO ERRORS, 1=INSUFFICIENT TIME,
C                                2=LOST SYNC, 3=ERROR COUNT OVERFLOW)
C       8:    LOW BYTE: PATTERN (CODE)
C       8:    HIGH BYTE: BLOCK SIZE (CODE)
      COMMON/COMFLG/ICOND,IEAS,JERR,NEWFIL
      COMMON/ARGS/ISKP(20),IERR(2)
      REAL*4 BITS,RATE
      INTEGER IBUF(256),RECORD(8,32)
      INTEGER IBITS(2),IRATE(2)
      INTEGER PATN,SIZE
      BYTE SYSNAM(6)
      EQUIVALENCE(IBUF(1),RECORD(1,1))
      EQUIVALENCE(IBITS,DBITS),(DRATE,IRATE)
      DATA NWDS/256/
C       GET SYSNAM OF BERC FILE
      CALL FILOC(2,3,'BERC.FIL',SYSNAM)
      IF(ICOND.EQ.0) GO TO 30
      IEAS=IEAS+1
      IF(ICOND.EQ.2) GO TO 10
C       DIRECTORY SEARCH ERROR
    5 CONTINUE
      IERR(2)=400
      GO TO 300
C       TRY FADE FILE
   10 CONTINUE
      NEWFIL=1
      CALL FILOC(1,3,'BERC.FIL',SYSNAM)
      IF(ICOND.EQ.0) GO TO 20
      IEAS=IEAS+1
      IF(ICOND.NE.3) GO TO 5
C       TRY HOP FILE
      NEWFIL=3
      CALL FILOC(1,3,'BERC.FIL',SYSNAM)
      IF(ICOND.EQ.0) GO TO 20
      IEAS=IEAS+1
      IF(ICOND.NE.3) GO TO 5
      GO TO 200
C       CREATE NEW BERC FILE
   20 CONTINUE
      IBUF(256)=0
      CALL SETFIL(3,SYSNAM,JERR,'DK ',1)
      GO TO 100
```

```
C       READ IN OLD BERC FILE
 30 CONTINUE
      CALL SETFIL(3,SYSNAM,JERR,'DK ',1)
      DO 50 IBLK=0,7
      CALL READRW(3,IBLK,IBUF,NWDS,JERR)
      IF(IBUF(256).LT.31) GO TO 100
 50 CONTINUE
C       BERC FILE FULL
      IERR(2)=600
      GO TO 300
C       ENTER NEW RECORD INTO BERC FILE
100 CONTINUE
      NREC=IBUF(256)+1
      OBITS=BITS
      ORATE=RATE
      RECORD(1,NREC)=NBLK
      RECORD(2,NREC)=IBITS(1)
      RECORD(3,NREC)=IBITS(2)
      RECORD(4,NREC)=ICOUNT
      RECORD(5,NREC)=IRATE(1)
      RECORD(6,NREC)=IRATE(2)
      CALL PACK(8,0,NBERC,RECORD(7,NREC))
      CALL PACK(8,8,ISTAT,RECORD(7,NREC))
      CALL PACK(8,0,PATN,RECORD(8,NREC))
      CALL PACK(8,8,SIZE,RECORD(8,NREC))
      IBUF(256)=NREC
      CALL WRITRW(3,IBLK,IBUF,NWDS,JERR)
      IF(NREC.NE.31) GO TO 150
      IBLK=IBLK+1
      IF(IBLK.EQ.8) GO TO 150
C       SET NUMBER OF RECORDS IN NEXT BLOCK TO 0
      CALL READRW(3,IBLK,IBUF,NWDS,JERR)
      IBUF(256)=0
      CALL WRITRW(3,IBLK,IBUF,NWDS,JERR)
150 CONTINUE
      ENDFILE 3
      RETURN
C       NO ROOM FOR BERC FILE
200 CONTINUE
      IERR(2)=500
300 CONTINUE
      IERR(1)=4
      RETURN
      END

ROUTINES CALLED:
FILOC , SETFIL, READRW, PACK  , WRITRW

SWITCHES = /ON,/SU,/CO

BLOCK        LENGTH
BFILE   799     (003076)*
COMFLG  4       (000010)
ARGS    22      (000054)

**COMPILER ----- CORE**
  PHASE       USED  FREE
DECLARATIVES 01886 16780
EXECUTABLES  01628 16246
ASSEMBLY     01313 19478
```

Subroutine XMIT

XMIT outputs the word JDOUT to the UDC-11 module, OUTMOD. It then reads back the data from the input module, INMOD, and compares it with JDOUT. JSEL is used to select Control Word A or B if necessary:

JSEL = 0, No Select Necessary

JSEL = 1, Select A

JSEL = 2, Select B

XMIT returns JERR as the error code:

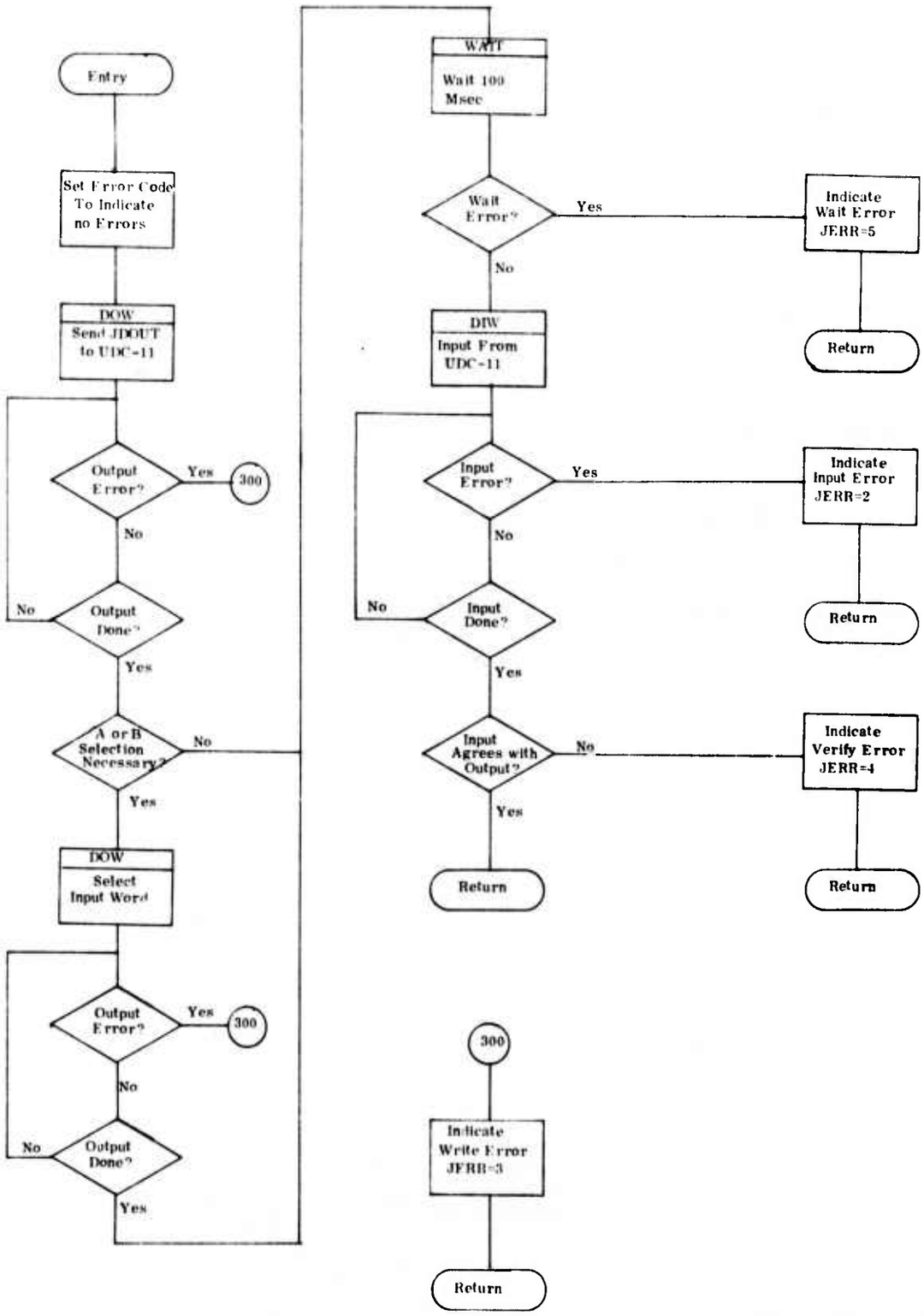
JERR = 1, No Errors

JERR = 2, Read Error

JERR = 3, Write Error

JERR = 4, Verify Error

JERR = 5, Wait Error


Entry
Set Error Code To Indicate no Errors
DOW
Send IDOUT to UDC-11
Output Error?
Yes
300
No
Output Done?
No
Yes
A or B Selection Necessary?
No
Yes
DOW
Select Input Word
Output Error?
Yes
300
No
Output Done?
No
Yes
WAIT
Wait 100 Msec
Wait Error?
Yes
Indicate Wait Error JERR=5
Return
No
DIW
Input From UDC-11
Input Error?
Yes
Indicate Input Error JERR=2
Return
No
Input Done?
No
Yes
Input Agrees with Output?
No
Indicate Verify Error JERR=4
Return
Yes
Return
300
Indicate Write Error JERR=3
Return

```
 FORTRAN V06.13                    09:18:03    11-DEC-74   PAGE    1

                      SUBROUTINE XMIT(OUTMOD,INMOD,JDOUT,JSEL,JERR)
                C       XMIT OUTPUTS JDOUT TO THE UDC-11, OUTMOD.  IT THEN
                C       READS BACK THE DATA FROM INMOD AND COMPARES IT WITH
                C       JDOUT.  JSEL IS USED TO SELECT CONTROL WORD A OR B:
                C       JSEL=0,NO SELECT NECESSARY
                C            =1, SELECT A
                C            =2, SELECT B
                C       JERR IS THE ERROR CODE,  1=NO ERRORS, 2=READ ERROR,
                C       3=WRITE ERROR, 4=VERIFY ERROR, AND 5=WAIT ERROR.
                      INTEGER OUTMOD,DGIS
                      DATA DGIS/'72/
                      JERR=1
                      CALL DDW(1,OUTMOD,JDOUT,JSTAT)
                   10 IF(JSTAT.EQ.3) GO TO 300
                      IF(JSTAT.EQ.2) GO TO 10
                      IF(JSEL.EQ.0) GO TO 20
                      JSEL=JSEL-1
                      CALL DDW(1,DGIS,JSEL,JSTAT)
                   15 IF(JSTAT.EQ.3) GO TO 300
                      IF(JSTAT.EQ.2) GO TO 15
                   20 CALL WAIT(100,1,JWFLG)
                      IF(JWFLG.NE.1) GO TO 500
                      CALL DIW(1,INMOD,JDIN,JSTAT)
                   30 IF(JSTAT.EQ.3) GO  TO 200
                      IF(JSTAT.EQ.2) GO TO 30
                      IF(JDOUT.NE.JDIN) GO TO 400
                      RETURN
                  200 JERR=2
                      RETURN
                  300 JERR=3
                      RETURN
                  400 JERR=4
                      RETURN
                  500 JERR=5
                      RETURN
                      END

                ROUTINES CALLED:
                DDW   , WAIT  , DIW

                OPTIONS =/ON,/SU,/CO,/OP:1

                BLOCK        LENGTH
                XMIT    194     (000624)*

                **COMPILER ----- CORE**
                   PHASE       USED  FREE
                DECLARATIVES  00982 02813
                EXECUTABLES   01223 01772
                ASSEMBLY      01058 06577
```

SECTION V

HARDWARE DESCRIPTION

This section describes the control and interface circuitry for the BERCs. For details concerning the BERCs, refer to the manufacturer's O&M manual.

1. INTRODUCTION

The BERC control logic provides the necessary timing, decoding, and buffering to properly select, under computer control, the desired parameters for a given test. In addition, counters and storage elements are provided to collect and transmit resultant test data to the CSEL computer.

For this discussion it is best to think of each BERC as containing two sections; a data generator (transmitter) and a meter (data analyzer). Each section is controlled by separate logic sections (see Figure 8).

Figures 9 and 10 show the basic data generator control elements for the 1210A and 3000, respectively. Figure 11 shows the basic meter control elements. This figure is applicable to both the 1210A and 3000 since their respective meter interfaces, although physically separate, operate essentially identically.

Optoisolators are used to separate the power supplies and signal grounds of the BERCs from that of the logic interfaces. This prevents ground loops, etc. from causing errors in control and/or data signals. The optoisolators are noninverting devices biased to be TTL compatible on both their inputs and outputs. See Figure 18, circuit U15, for the schematic of the optoisolators used in the BERC control logic.

2. DATA GENERATOR CONTROL

The data generator control interfaces are primarily static buffers for control words from the UDC-11 which initialize the data format in the BERCs prior to beginning a test. These interfaces also decode certain bits of these control words. In addition, status drivers are provided to retransmit the control words to the UDC-11 input registers for verification.

Figure 12 shows the overall data generator control cabling for the 1210A and 3000.

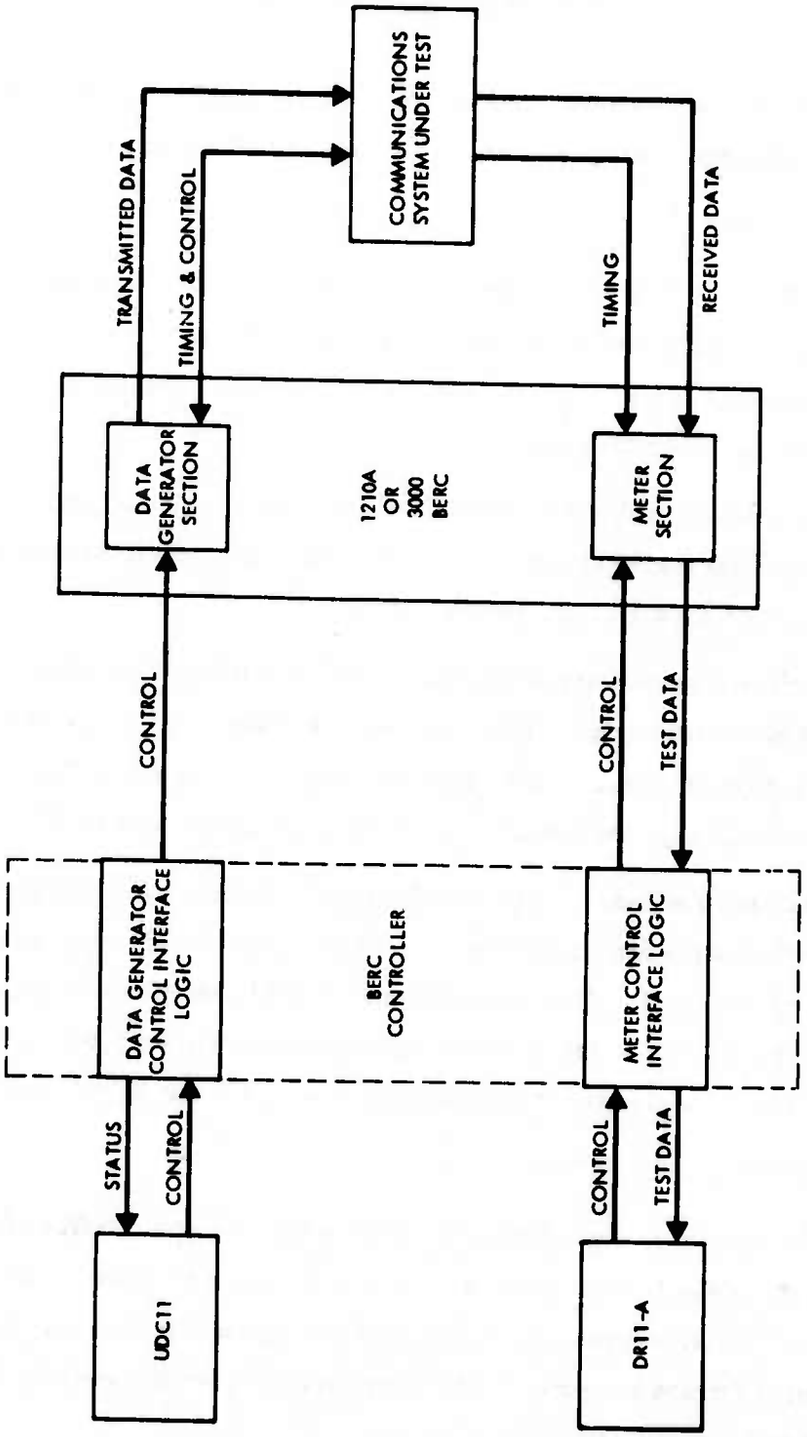


Figure 8. Overall Control Block Diagram

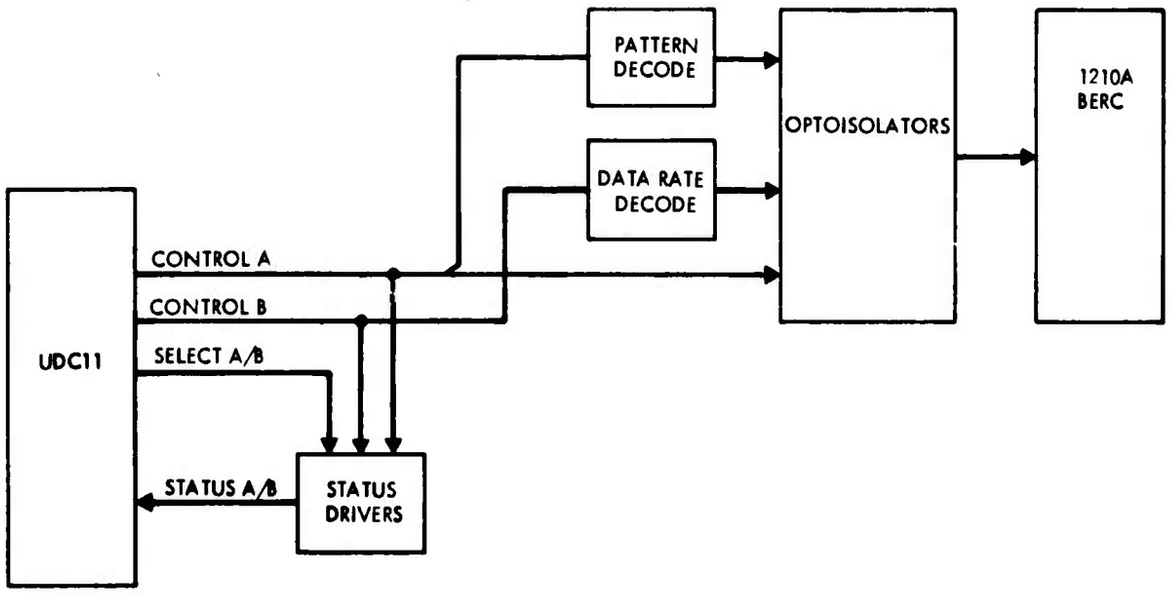


Figure 9. 1210A Data Generator Control Block Diagram

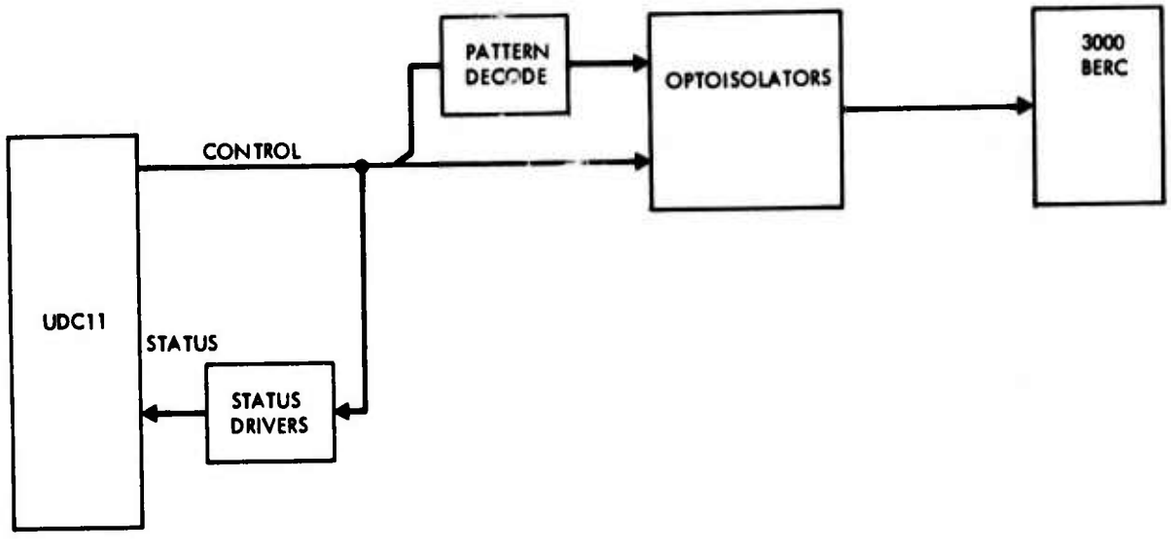


Figure 10. 3000 Data Generator Control, Block Diagram

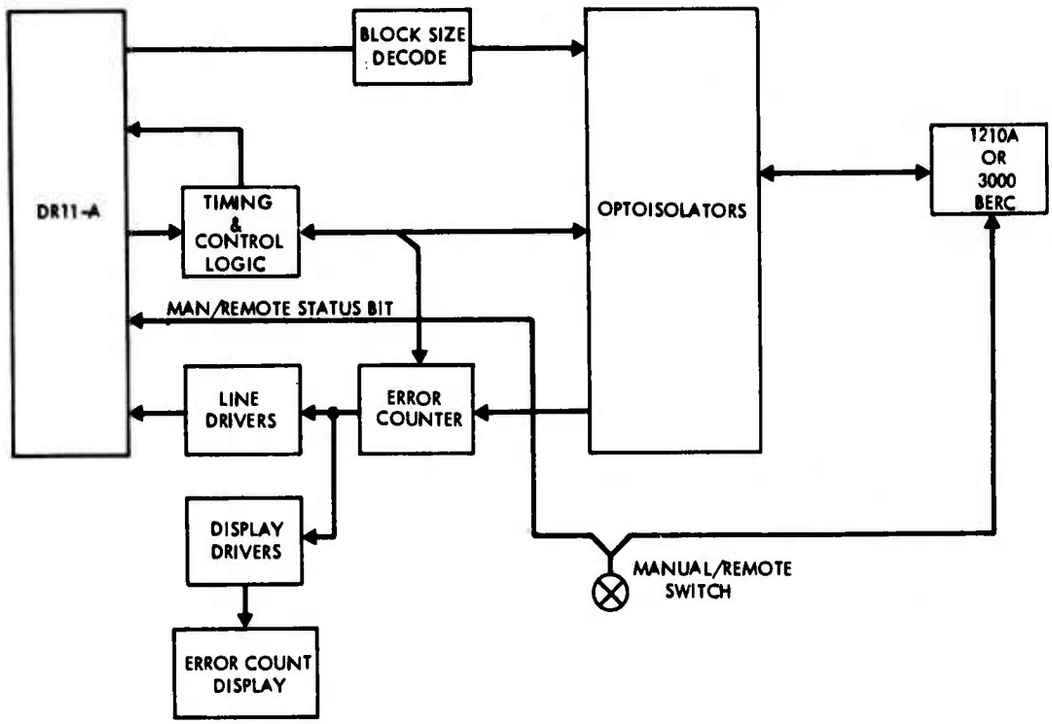


Figure 11. Meter Control, Block Diagram

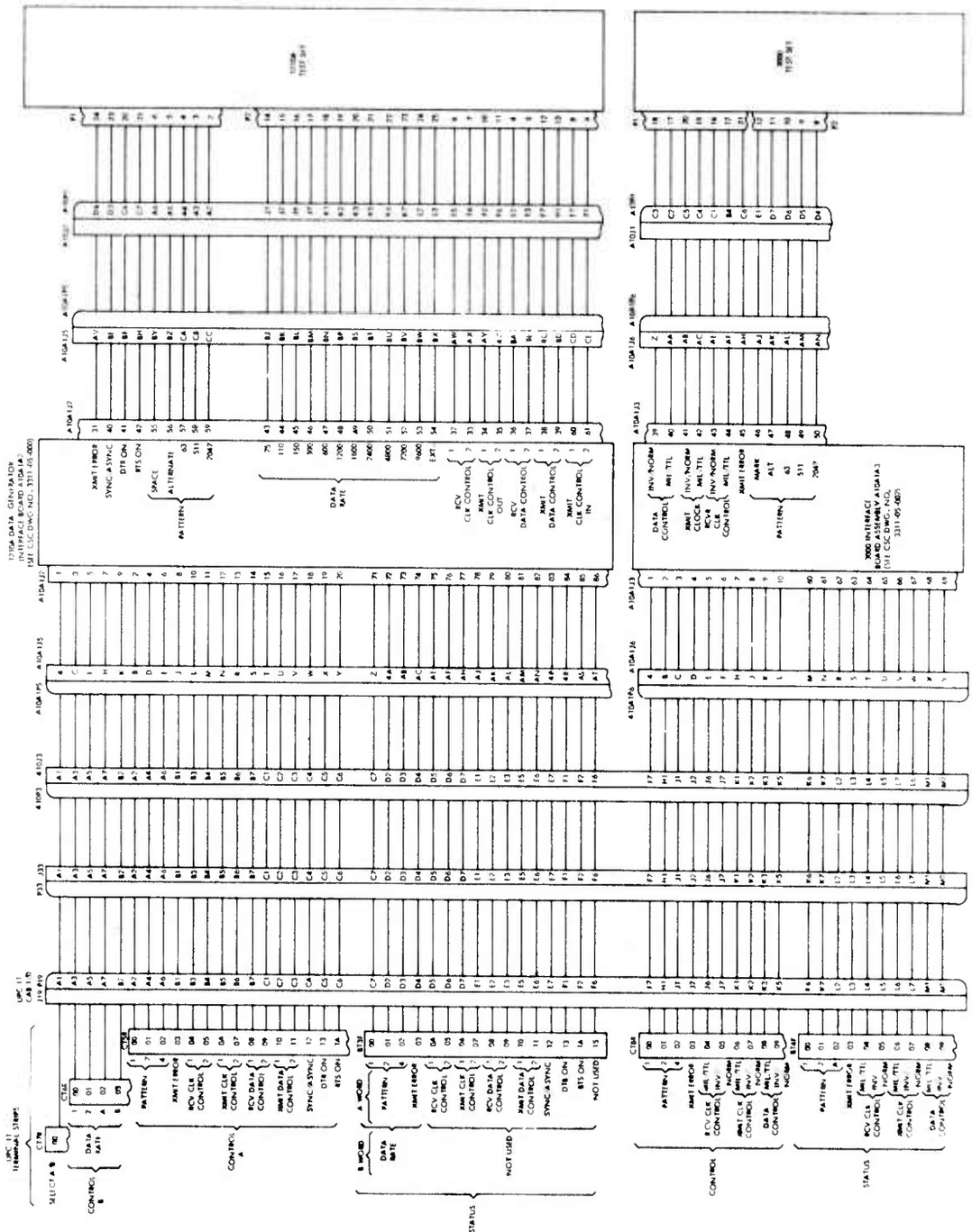


Figure 12. 3000 and 1210A Data Generator Control, Interface Cabling (3311-05-005)

In the UDC-11, the control registers are BM685 modules with BW403 signal conditioning modules. The status registers are BW731 modules with BW402 signal conditioners.

Both the control and status signals in the data generator interfaces are negative true. There are two reasons for this. The first is that both the 1210A and 3000 recognize 0 volt at their inputs as an active control signal. The second is that the UDC-11 inverts the logic levels in both the control and status registers. For further clarification see the instruction manual for the K-band terminal simulator Volume II Paragraph 13.2.1 prepared under Contract F33615-72-C-2187.

For the sake of brevity, all integrated circuits and connectors will be referred to only by their circuit symbol number, and the pin referred to will follow (i.e., integrated circuit U28 pin 3 will be noted as U28, 3 etc.). In addition, when the signal flow through a gate is discussed, the circuit symbol number will be listed, followed first by the input pin(s), followed by a dash then the output pin. For example, on Figure 17 with input pins 12 and 13, and output pin 11 will be written U7, 12, 13 - 11.

a. Model 1210A Data Generator Control Interface

Figures 13 and 14 show the details of the logic interface for the 1210A data generator control. Control signals enter through A10A1J2 and go to the inputs of optoisolators U5 through U20. The PATTERN bits are decoded by U28 (see Table 5) and the DATA RATE bits are decoded by U26 and U27 (see Table 6) before going to the optoisolator inputs. All control signals also go to the inputs of status input drivers U30 through U37. Resistor array U25 provides pullups on the control lines feeding the PATTERN and DATA RATE decoders, which gives increased noise immunity. Resistor networks U1 through U4 and U21 through U24 provide proper biasing for optoisolators U5 through U20.

Depending on the state of the SELECT A/B bit, status input drivers U30 through U37 retransmit either the CONTROL A or CONTROL B words to the UDC-11 status register for verification.

TABLE 5

TRUTH TABLE FOR 1210A PATTERN DECODE U28

UDC Code			U28 Input Line			U28 Output						Selected Pattern
						7	6	5	4	3	Pin	
4	2	1	4	2	1	1	2	3	4	5	Code	
0	0	0	1	1	1	1	1	1	1	1		Mark
0	0	1	1	1	0	0	1	1	1	1		Space
0	1	0	1	0	1	1	0	1	1	1		Alternate
0	1	1	1	0	0	1	1	0	1	1		63
1	0	0	0	1	1	1	1	1	0	1		511
1	0	1	0	1	0	1	1	1	1	0		2047

NOTE: On Output, 0 is Controlling State.

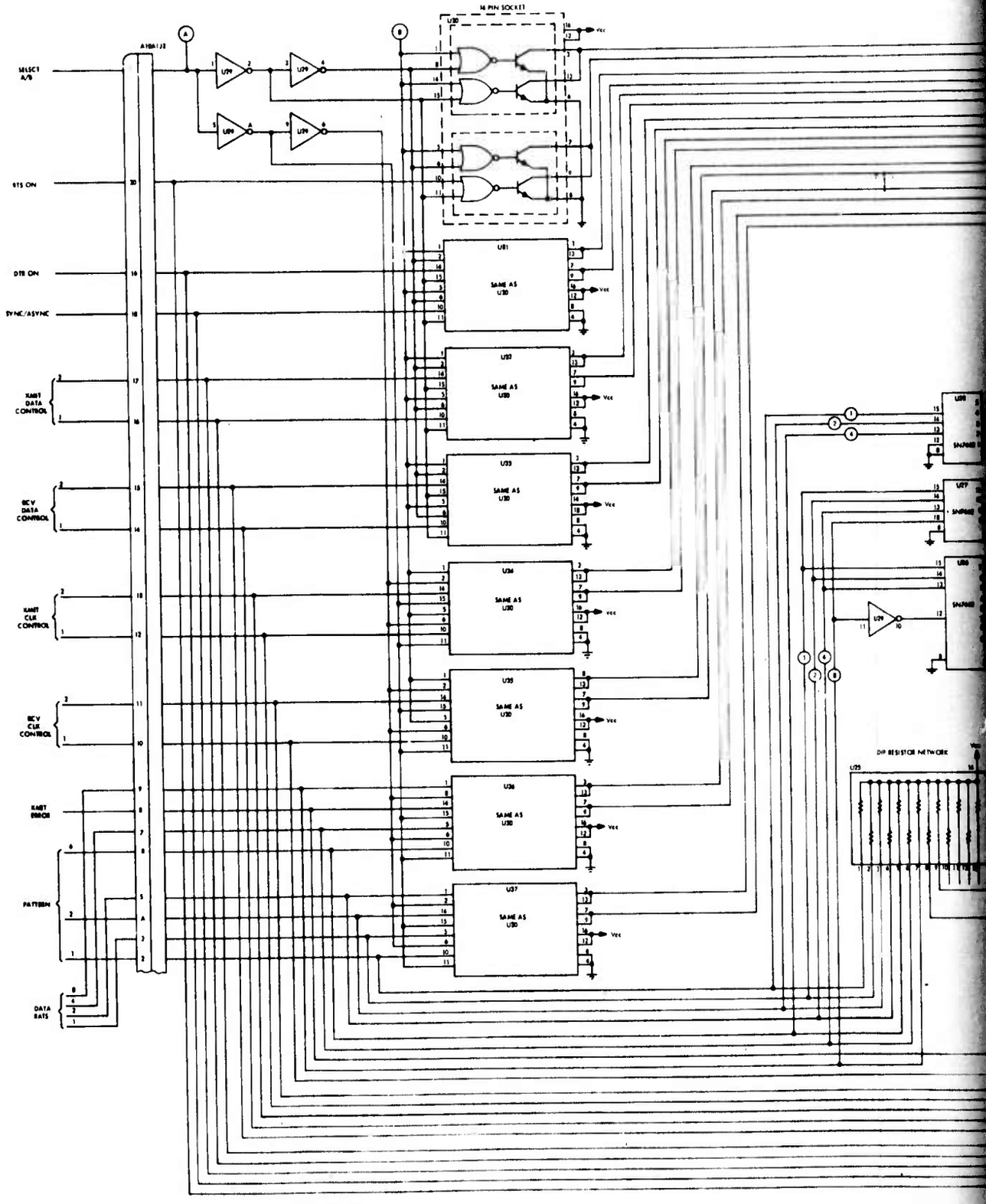


Figure 13. 1210A Ge
A10

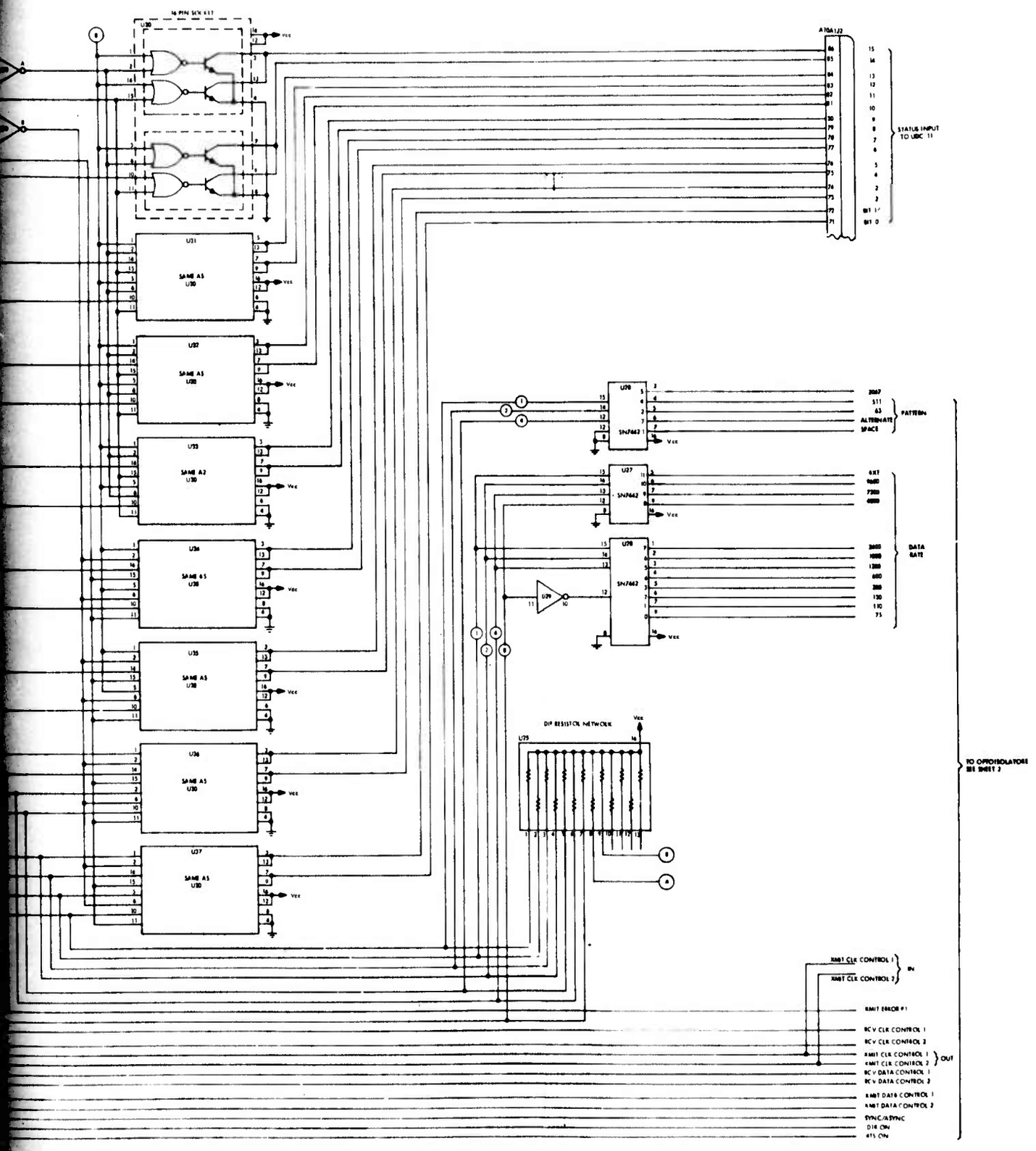


Figure 13. 1210A Generator Control Interface Board Assembly A10A1A2, Logic Diagram

TABLE 6

TRUTH TABLE FOR 1210A DATA RATE DECODE U26 AND U27

UDC Code				U26, U27, U29 Input Line				U26, U27 Output												Selected Data Rate
								U26						U27						
								9	7	6	5	4	3	2	1	9	7	6	5	
8	4	2	1	8	4	2	1	0	1	2	3	4	5	6	7	8	9	10	11	Pin Code
0	0	0	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	75
0	0	0	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	110
0	0	1	0	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	150
0	0	1	1	1	1	0	0	1	1	1	0	1	1	1	1	1	1	1	1	300
0	1	0	0	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	600
0	1	0	1	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1200
0	1	1	0	1	0	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1800
0	1	1	1	1	0	0	0	1	1	1	1	1	1	1	0	1	1	1	1	2400
1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	4800
1	0	0	1	0	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1	7200
1	0	1	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	9600
1	0	1	1	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	0	Ext.

NOTE: On Output, 0 is Controlling State.

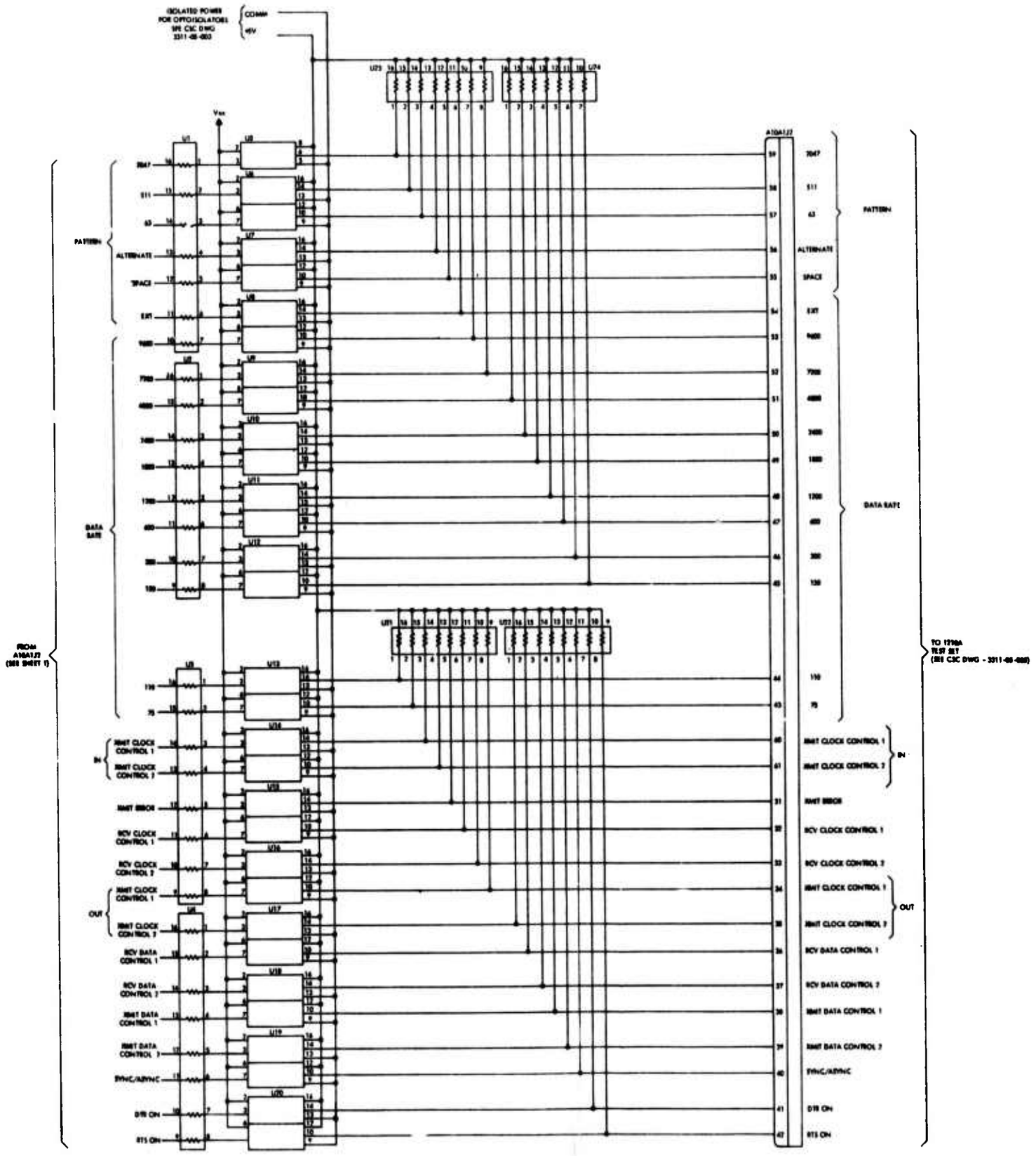


Figure 14. 1210A Data Generator Control Interface Board Logic Diagram Assembly A10A1A2

In the active state, the drivers furnish a return to relay coils in the BW731 modules in the UDC-11. Since these relays are powered by 6 Vdc, the driver outputs will be at 0 Vdc in the "on" (conducting) state, and 6 Vdc in the "off" (nonconducting) state.

Assume that CONTROL A word is selected. The SELECT A/B line J2,1 is a '1'. Inverters U29, 2 and U29, 6 then send a '0' to pins, 11 and 15 of the status drivers enabling the CONTROL A inputs pins 10 and 14. At the same time, inverters U29, 4 and U29, 8 send a '1' to pins 2 and 6 of the status drivers disabling the CONTROL B inputs, pins 1 and 5. If CONTROL B were selected, all the conditions would simply reverse.

On the output side of the status drivers, pins 3 and 13 form a wired OR as do pins 7 and 9.

Notice that the status input drivers retransmit the selected control word non-inverted.

On CONTROL A word, bit 15 is not used, and on CONTROL B word bits 4 through 15 are not used. The status driver input pins corresponding to these bits are put at logic '1' by connecting them to U25, 9 causing the computer to read these as "zeros".

b. Model 3000 Data Generator Control Interface

Figure 15 shows the details of the logic interface for the 3000 data generator control. Except for being simpler, this interface operates in much the same manner as the one used with the 1210A.

Control signals enter on A10A1J3 and go to the inputs of the status input drivers U1, U2, and U3. These signals are also connected to the inputs of optoisolators U31 through U36. The PATTERN control bits are decoded by U4 (see Table 7) before going to the optoisolator inputs. U17, U18, U26 and U27 provide proper biasing for the optoisolators. Part of U18 is also used to provide pullups at the input of decoder U4.

TABLE 7

TRUTH TABLE FOR 3000 PATTERN DECODE U4

UDC Code			U4 Input Line			U4 Output						Selected Pattern
						7	6	5	4	3	Pin	
4	2	1	4	2	1	1	2	3	4	5	Code	
0	0	0	1	1	1	1	1	1	1	1		Space
0	0	1	1	1	0	0	1	1	1	1		Mark
0	1	0	1	0	1	1	0	1	1	1		Alternate
0	1	1	1	0	0	1	1	0	1	1		63
1	0	0	0	1	1	1	1	1	0	1		511
1	0	1	0	1	0	1	1	1	1	0		2047

NOTE: On Output, 0 is Controlling State.

In this interface there is no multiplexing of the control bits at the status driver inputs, and a positive voltage from U18,5 is used at the enable inputs.

3. METER CONTROL

As shown in Figure 11, the meter control interface logic employed with each BERC handles all communications between the BERC and its associated DR11-A registers. Except where noted, all logic levels utilized in the meter control interfaces are TTL, positive true.

The meter control interface for each BERC contains the following:

1. Logic required for proper sequencing of control signals to BERC, and status signals to DR11-A.
2. A decode for BLOCK SIZE bits.
3. A 12-bit binary counter which totalizes errors which occur in a given test. Logic storage is provided to show when counter has overflowed at least once.
4. Buffers to transmit error count to DR11-A.
5. Drivers for front panel monitor lamps.

a. 1210A Meter Control Interface

Figure 16 shows the meter control interface cabling for the 1210A as well as the connection for the 1210A optoisolator power supply A10PS2.

Figures 17 and 18 show the details of the meter control interface logic for the 1210A. In Figure 17, MAN/REMOTE switch A10S1 is shown in the REMOTE position. The lower section puts the MAN/REMOTE flag to ground (logic 0) - the upper section puts the common of the optoisolator power supply on the remote control line to the 1210A which puts it in the remote mode. If A10S1 were placed in the MANUAL position,the MAN/REMOTE flag would be put at a 1 by the pullup at U9,3. The remote control line would be put at a 1 by the pullup at U22,2. This would put the 1210A in the manual mode.

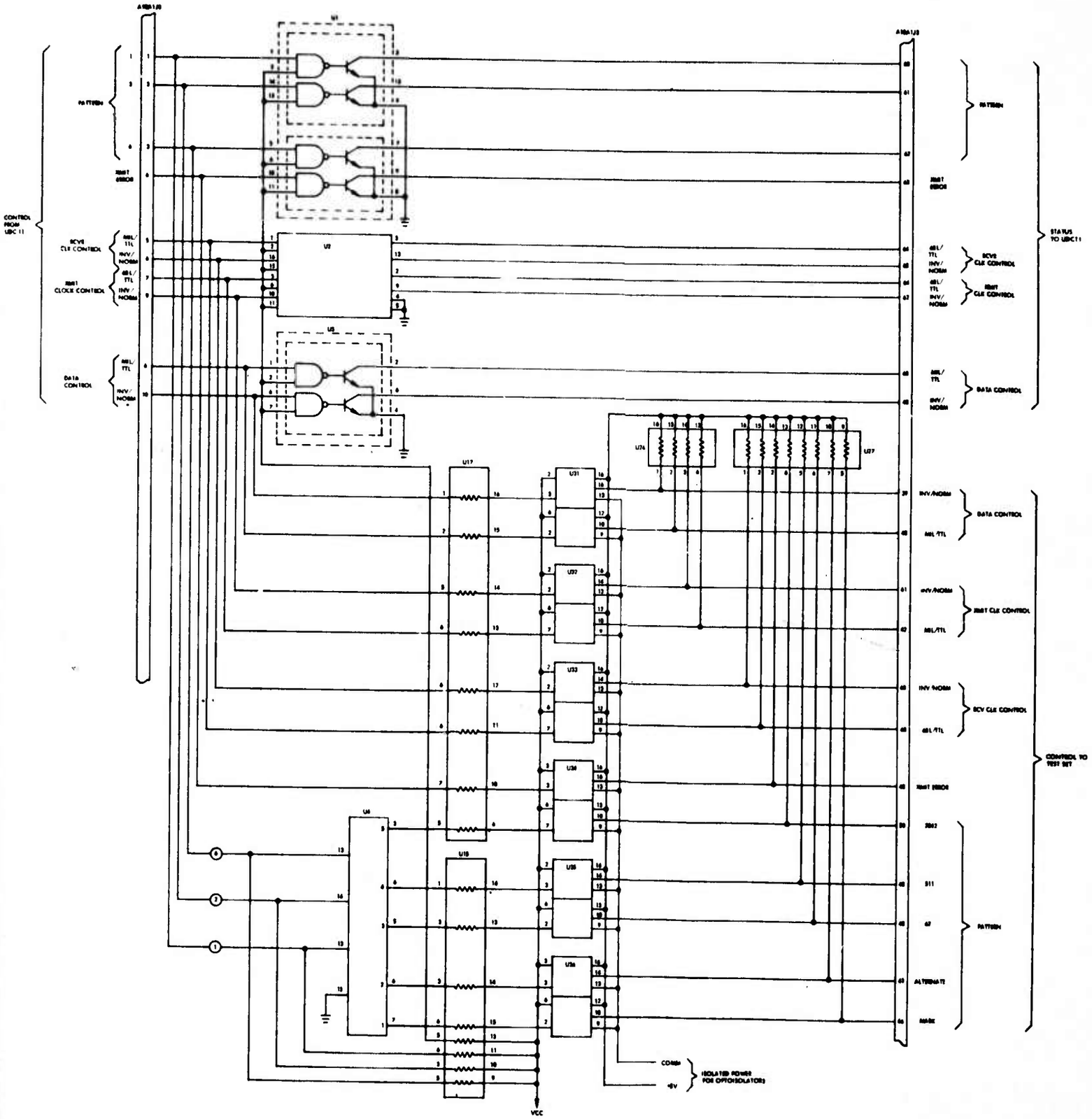
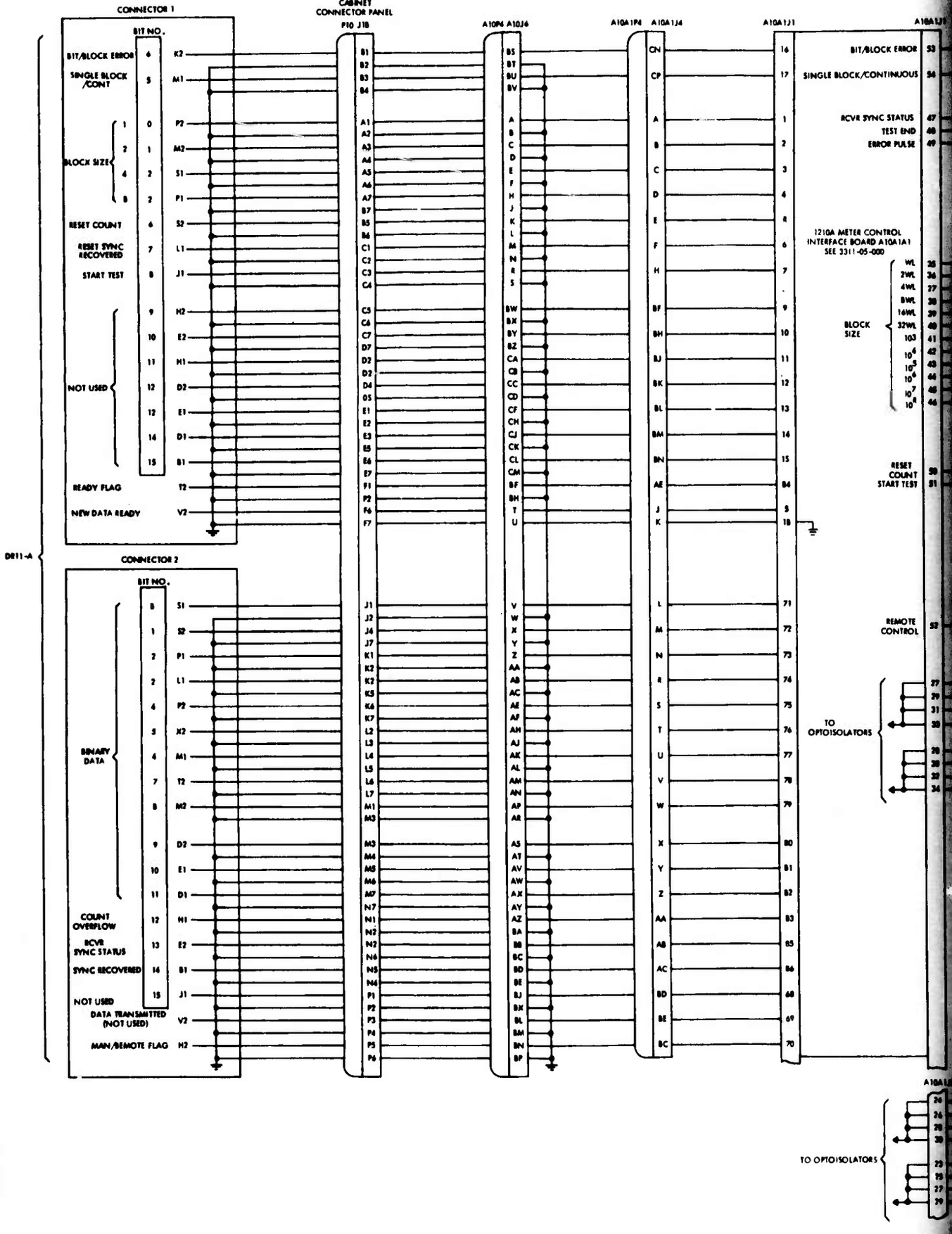


Figure 15. 3000 Interface Board, Data Generator Section


CONNECTOR 1
BIT NO.
BIT/BLOCK ERROR
SINGLE BLOCK /CONT
BLOCK SIZE
RESET COUNT
RESET SYNC RECOVERED
START TEST
NOT USED
READY FLAG
NEW DATA READY
CONNECTOR 2
BINARY DATA
COUNT OVERFLOW
RCVR SYNC STATUS
SYNC RECOVERED
NOT USED
DATA TRANSMITTED (NOT USED)
MAN/REMOTE FLAG
D011-A
CABINET CONNECTOR PANEL
P10 J1B
A10P4 A10J6
A10A1P4 A10A1J4
A10A1J1
A10A1J2
BIT/BLOCK ERROR
SINGLE BLOCK/CONTINUOUS
RCVR SYNC STATUS
TEST END
ERROR PULSE
1210A METER CONTROL INTERFACE BOARD A10A1A1 SEE 3311-05-000
BLOCK SIZE
RESET COUNT START TEST
REMOTE CONTROL
TO OPTOISOLATORS
TO OPTOISOLATORS

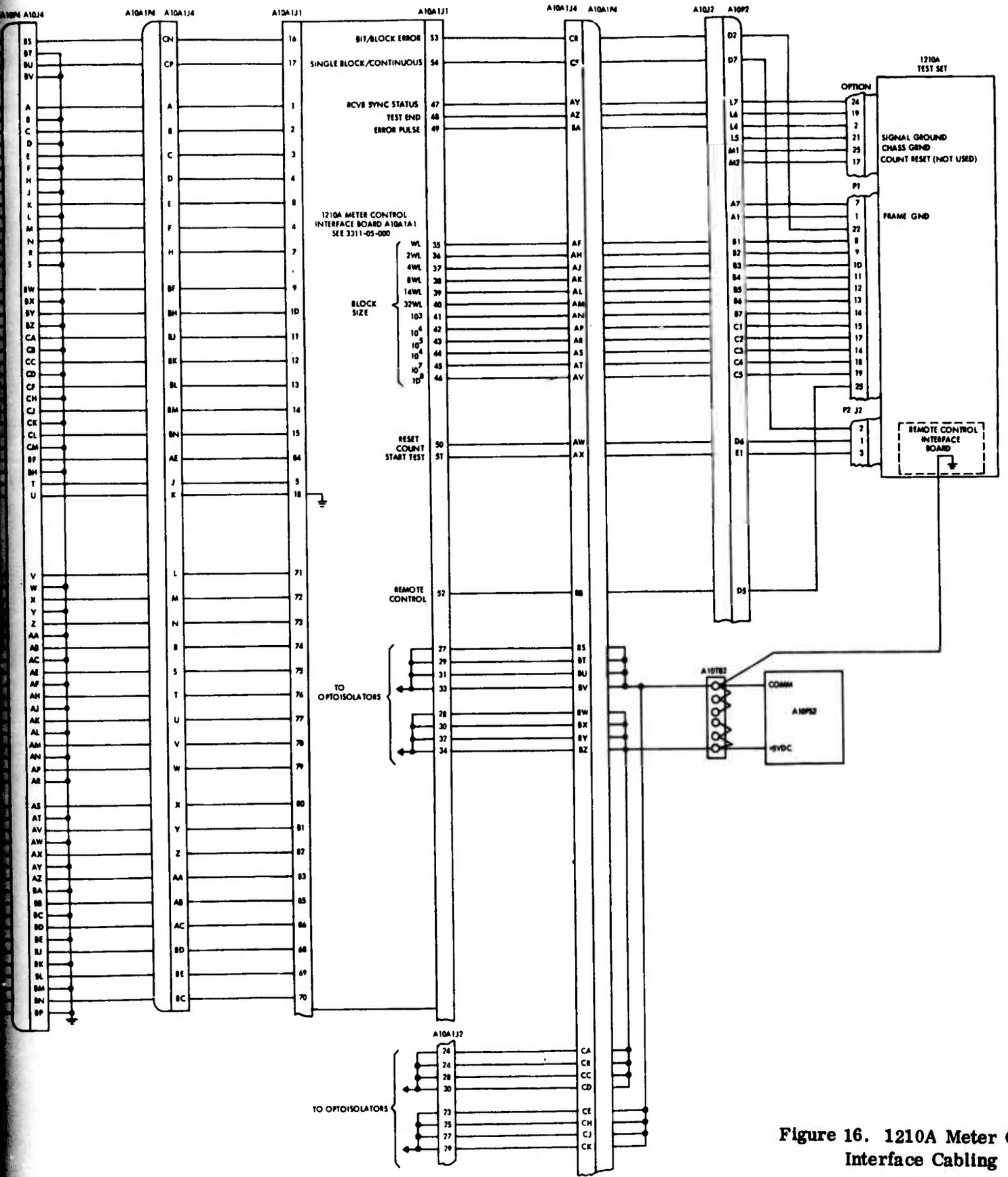
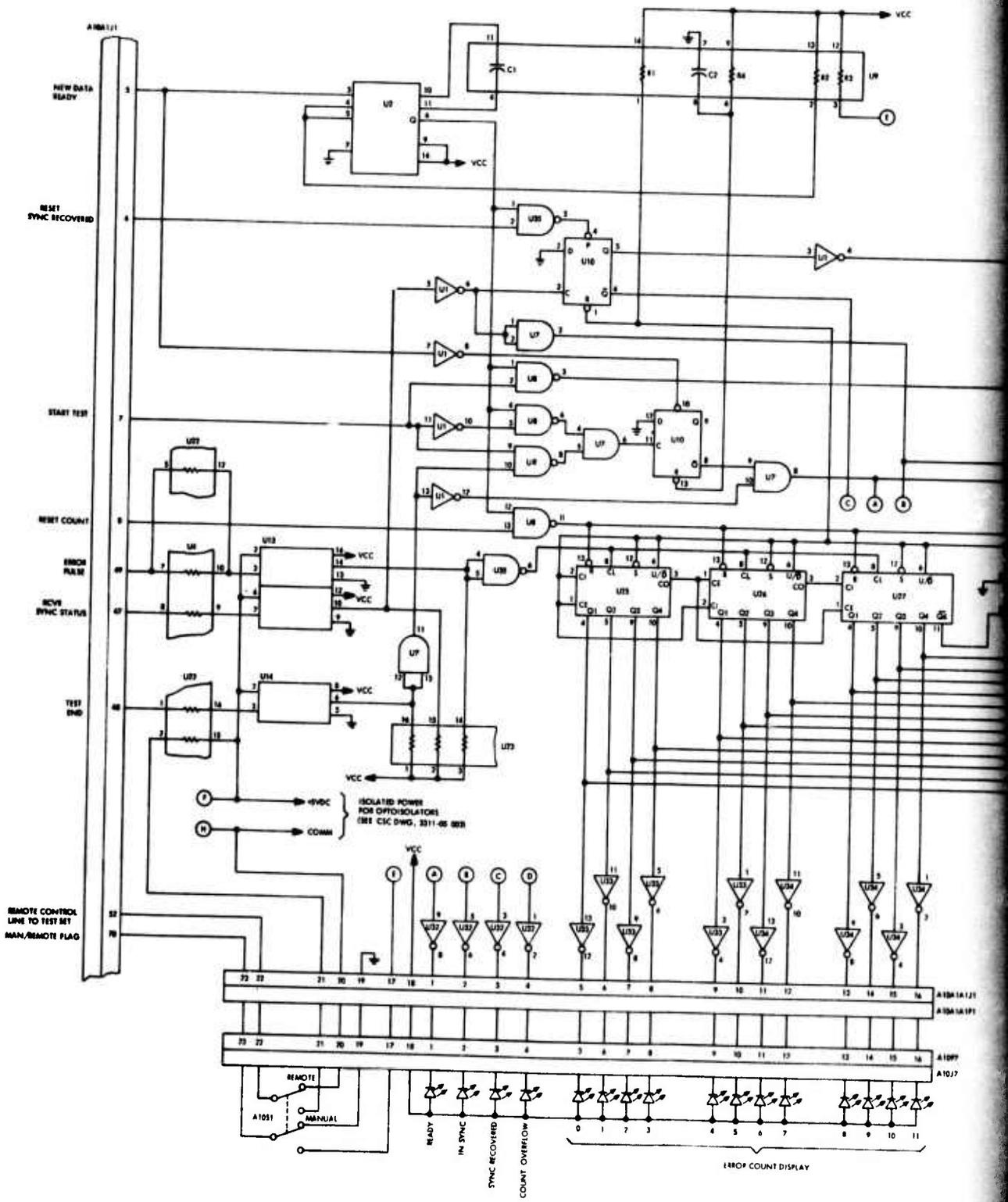


Figure 16. 1210A Meter Control, Interface Cabling

67/68


NEW DATA READY
RESET SYNC RECOVERED
START TEST
RESET COUNT
ERROR FLAG
RCVR SYNC STATUS
TEST END
REMOTE CONTROL LINE TO TEST SET
MAN/REMOTE FLAG
VCC
+5VDC
COMM
ISOLATED POWER FOR OPTOISOLATORS
REMOTE
MANUAL
A1051
READY
IN SYNC
SYNC RECOVERED
COUNT OVERFLOW
ERROR COUNT DISPLAY

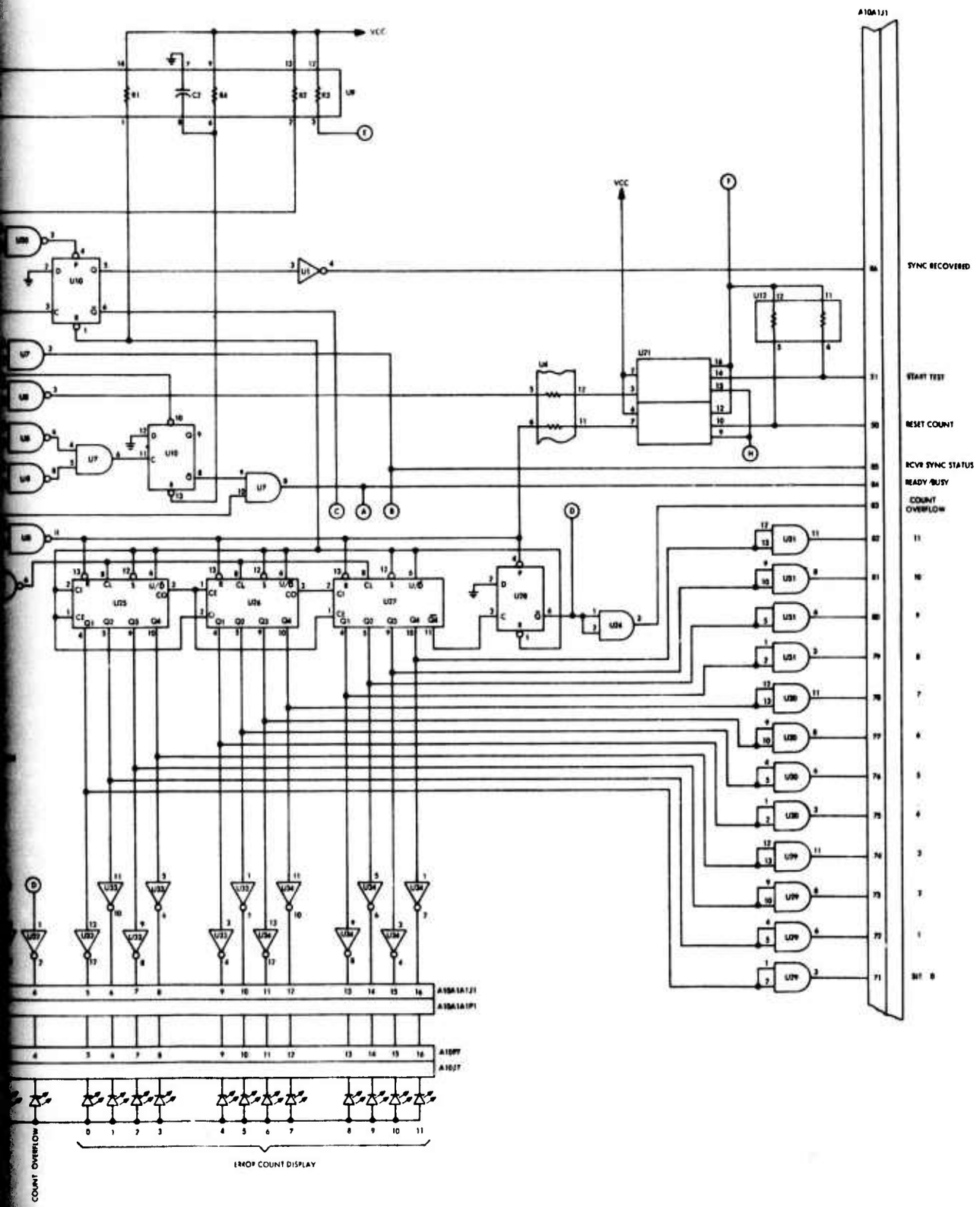


Figure 17. 1210A Meter Control Interface Board Assembly A10A1A1 Logic Diagram, Interface and Count Logic

2

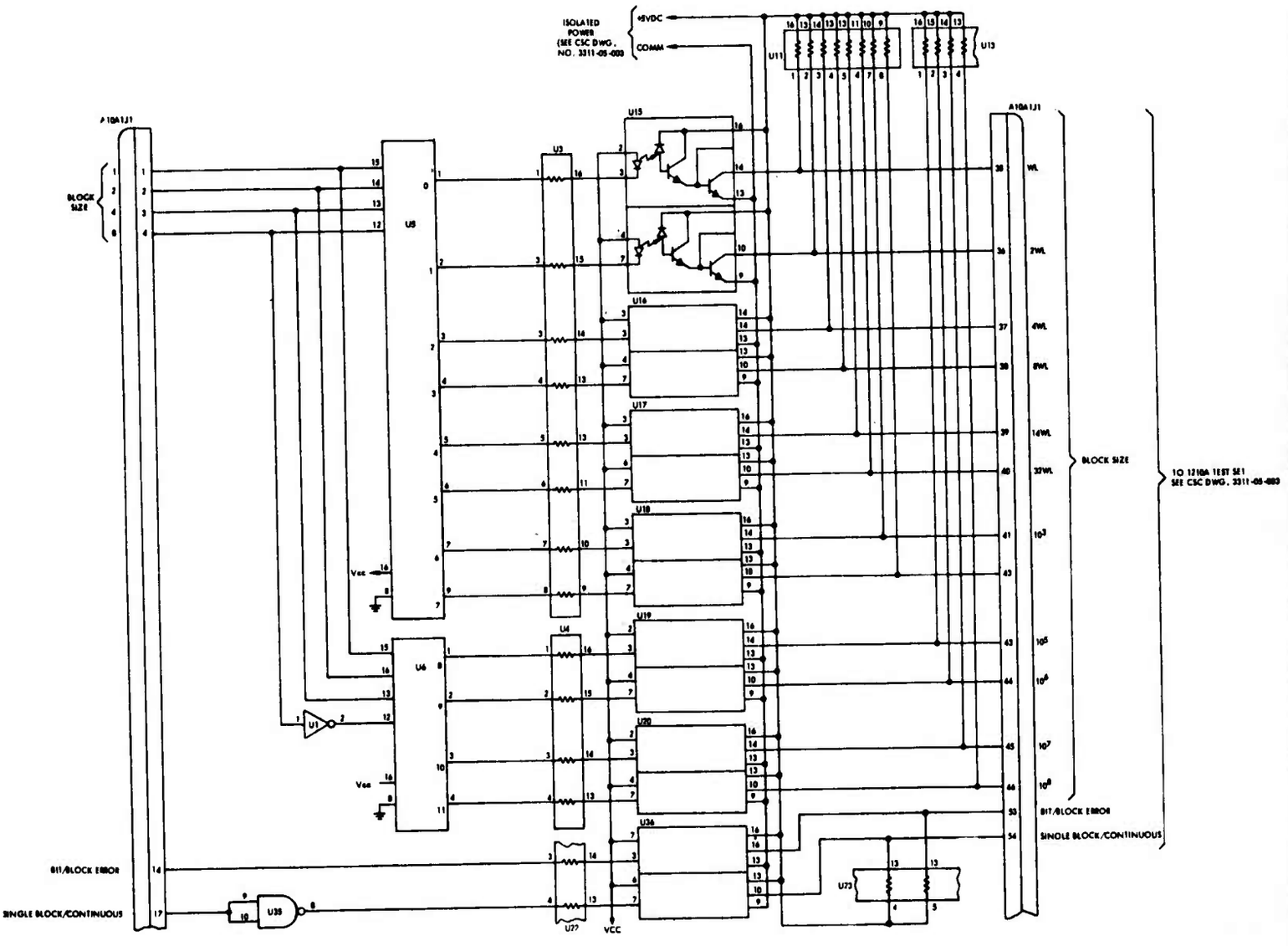


Figure 18. 1210A Meter Control Interface Board Assembly A10A1A1 Logic Diagram, Block Size Decode Section

As previously stated, the PDP-11 computer running CSEL, before starting a test, checks the MAN/REMOTE flag to ensure that the BERC is in the remote mode. When a test is to be started, the PDP-11 loads the COMMAND WORD register in the DR11-A (see Table 7). The gating and control logic in the DR11-A then issues a NEW DATA READY pulse which is a positive pulse, approximately 500 nanoseconds wide.

The NEW DATA READY pulse goes to U1, 9 and U2, 3. It is inverted at U1, 8 and sent to the preset input of the READY/BUSY section of flip flop U10, 10. The preset input of the flip flops used in this interface are level controlled with the 0 state active. The preset input U10, 10 is thus activated immediately upon receipt of the inverted NEW DATA READY pulse. Figure 19 gives the timing relationships of the events to be discussed next. Presetting the READY/BUSY section of U10 causes U10, 8 to put a 0 on the input of AND gate U7, 9. This causes U7, 8 to put a 0 on the READY/BUSY line. The interface is thus set to a BUSY state immediately upon receipt of the NEW DATA READY pulse. U7, 8 also connects to U32, 9, so when U7, 8 goes to 0 (BUSY), lamp driver U32, 9-8 is disabled and the READY lamp on the front panel is extinguished.

One shot U2 initiates and separates events so that they occur in proper order. The output of U2, 6 is a positive pulse approximately 20 microseconds wide. This pulse width is set by the capacitor mounted on pins 4 and 11 of component holder U9, and connected across pins 10 and 11 of U2. A 1K pullup resistor on U9, 2 is connected to U2, 4, and 5. This enables the one shot input U2, 3.

If the BERC had detected a loss and recovery of sync during a test, on the SYNC RECOVERED flip flop (U10), pin 5 would be a 0 and pin 6 would be a 1. U1, 4 would then be a 1 asserting the SYNC RECOVERED bit. The 1 on U10, 6 would enable the 3-4 section of lamp driver U32, so the SYNC RECOVERED light on the front panel would be on. If the RESET SYNC RECOVERED line at U35, 2 is asserted when U2 "fires," the positive pulse from U2, 6 will be inverted at U35, 3 and sent to the preset input pin 4 of the SYNC RECOVERED section of dual flip flop U10. This now causes U10, 5 to go to 1

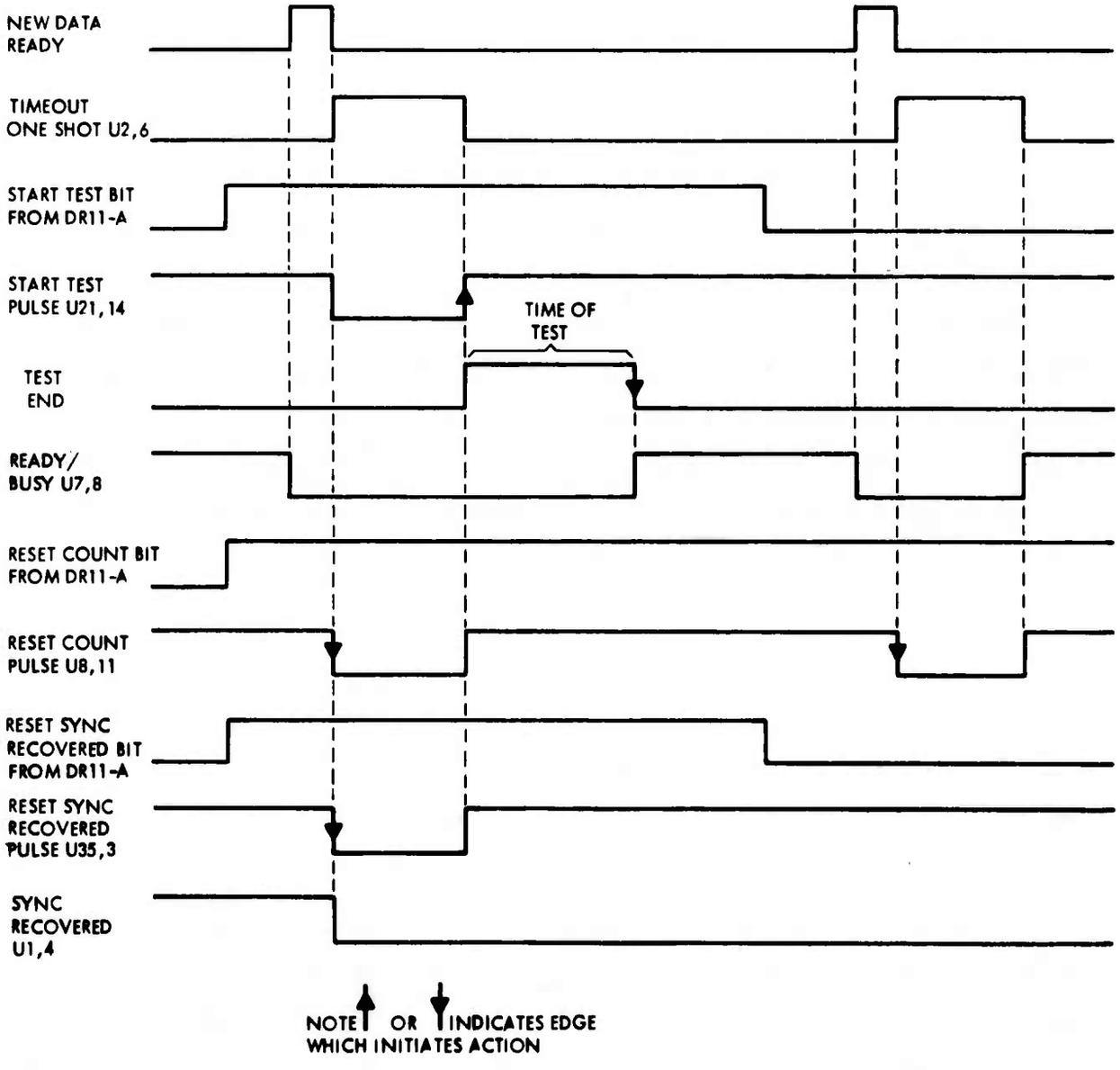


Figure 19. 1210A Meter Control Interface, Timing Diagram

and U10,6 to go to 0. With U10, 5 a 1, U1,4 is now 0 so the SYNC RECOVERED bit is no longer asserted. With U10,6 a 0, the 3-4 section of lamp driver U32 is now disabled so the SYNC RECOVERED lamp is now extinguished.

If the RESET COUNT line at U8,13 is high, the positive pulse from one shot U2,6 is inverted at U8,11 and sent to the reset inputs (pin 13) of the binary error counters U25, U26, and U27 clearing them. This RESET COUNT pulse from U8,11 also goes to the preset input of the COUNT OVERFLOW flip flop U28,4. If a count overflow had been previously detected, U28,6 would be a 1. The RESET COUNT pulse would cause U28,6 to go to 0 which is transmitted by buffer U24 to the COUNT OVERFLOW bit in the DR11-A. U28,6 also connects to the 1-2 section of lamp driver U32, so the 0 now on U28,6 disables this driver and extinguishes the COUNT OVERFLOW lamp. Finally the RESET COUNT pulse goes to the input of optoisolator U21,7. The output, U21,10 sends the RESET COUNT pulse to the BERC which resets the error counter and overflow indicator.

If the START TEST line at U8,2 is asserted, the pulse from U2,6 will be inverted at U8,3 and sent to the input of optoisolator U21,3. The output U21,14 sends the start test pulse to the BERC. Note from Figure 19 that the test is started by the trailing edge of the START TEST pulse, so that all counters and status flip flops will have been initialized prior to the beginning of a test.

At this point some logic levels should be noted. Remember that the START TEST command line was asserted so it is a 1. This 1 is seen at U8,9 so the TEST END signal on U8,10 will be seen inverted at U8,8 and U7,5. The START TEST command, inverted by U1,1-10, is seen as a 0 at U8,5 so the one shot pulse at U8,4 is ignored and U8,6 and U7,4 are now a constant 1. Since U7,4 is a 1, the transitions at U7,5 can now be coupled to the clock input of the READY/BUSY flip flop U10,11.

When the BERC senses the trailing edge of the START TEST pulse, it puts the TEST END line to 1, which indicates that a test is in progress. This 1 is coupled through optoisolator U14,3-6 and buffer U7,12, 13-11 to U1,13 and U8,10. This 0 to

1 transition is inverted at U8,8 and coupled through U7,5-6 to the READY/BUSY flip flop clock input, U10,11 as a 1 to 0 transition. Since the flip flops used in this interface are clocked by a rising edge (0 to 1) this 1 to 0 transition is ignored.

When the BERC has finished running the test as programmed, it puts the TEST END line from 1 to 0. This is inverted by U8,10-8 and coupled by U7,5-6 to the clock input U10,11 as a 0 to 1 transition. This transition toggles the flip flop, which causes U10,8 to go to 1. U7,9 and 10 are now 1, so U7,8 goes to 1 which indicates a READY condition to the processor. At the same time, lamp driver U32,9-8 is now enabled and the READY lamp is lit.

Assume that the START TEST line has not been asserted. U8,2 will be a 0, so the one shot pulse at U8,1 will be ignored and a start test pulse will not be issued. U1,11 is a 0 which is inverted to a 1 at U1,10 and sent to U8,5. The positive pulse from one shot U2 can now be inverted by U8,4-6 and sent to U7,4. Since U8,9 is a 0, U8,8 is a constant 1 and so U7,5 is also. This means that U7,4-6 can couple the negative going pulse from U8,6 to the clock input U10,11. Again, the falling edge is ignored and the rising edge toggles the flip flop which, as previously stated, puts a 1 on U7,9.

A word about U1,13-12. If the BERC is not running a test, U1,13 will be a 0 which is inverted to a 1 at U1,12 and sent to U7,10, which enables U7,9-8 to pass whichever state is at U10,8. If however the BERC were in the CONTINUOUS mode, the TEST END line would be continuously high. This would put a constant 1 at U1,13 which would put a constant 0 at U7,10. U7,8 would thus transmit a 0 (BUSY) regardless of the state of READY/BUSY flip flop U10,8.

The BERC transmits one error pulse each time an error is detected. These error pulses are coupled through optoisolator U13,3-14, buffered by U35,4,5-6 and sent to the block inputs pin 8 of binary error counters U25, U26, and U27. U33 and U34 drive the ERROR COUNT display lamps on the front panel. U28,U30 and U31 transmit the error count to the DR11-A DATA WORD register.

As previously stated, the error counter in the interface logic overflows after a count of 4095. On the 4096th count, U27,11 goes from 0 to 1 which toggles the COUNT OVERFLOW flip flop U28, putting U28,6 to 1. Once set, U28 ignores all other inputs at its clock input until present again by the COUNT RESET pulse. U24,1,2-3 drives the COUNT OVERFLOW line to the DR11-A.

The RCVR SYNC STATUS signal from the BERC is coupled through optoisolator U13,7-10 and inverter U1,5-6 to the clock input of the SYNC RECOVERED flip flop U10,3. Buffer U7, 1,2-3 drives the RCVR SYNC STATUS line to the DR11-A and the input of IN SYNC lamp driver U32,5-6.

The the BERC receiver is in sync, the RCVR SYNC STATUS line from the BERC is a 0. This is inverted at U1,6. Buffer U7,1,2-3 puts a 1 on the RCVR SYNC STATUS line to the DR11-A and to U32,5 so the IN SYNC lamp is now lit. If the BERC should detect a loss of sync, the RCVR SYNC STATUS line will go to 1. Inverter U1,5-6 now puts a 0, through buffer U7,1,2-3 to the DR11-A RCVR SYNC STATUS line, indicating a loss of sync. U32,5 now also is a 0 so the IN SYNC lamp is extinguished.

When the BERC recovers sync,the RCVR SYNC STATUS line again falls to 0. At U1,6 this is seen as a 0 to 1 transition. This transition at U10,3 toggles the SYNC RECOVERED flip flop which puts a 0 at U10,5 and a 1 at U10,6. Inverter U1,4 now becomes 1 so the SYNC RECOVERED bit is asserted. The 1 at U10,6 enables lamp driver U32,3-4 so the SYNC RECOVERED lamp is now lit.

Refer to Figure 18. The BIT/BLOCK ERROR control bit is coupled through optoisolator U36,3-14 to the BERC. The SINGLE BLOCK/CONTINUOUS bits is coupled through inverter U35,9,10-8 then through optoisolator U36,7-10 to the BERC. Refer to Table 2 for an explanation of the function of these bits.

U5, U6 and U1, 1-2 form a decode for the BLOCK SIZE bits (see Table 8). Optoisolators U15 through U20 couple the decoded BLOCK SIZE bits to the 1210A BERC.

Resistor networks U3, U4, U11, U12, U22, and U23 provide biasing for the Optoisolators.

TABLE 8

TRUTH TABLE FOR 1210A BLOCK SIZE DECODE U5 AND U6

	INPUT				OUTPUT											
LINE	8	4	2	1	0	1	2	3	4	5	6	7	8	9	10	11	
	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	WL
	0	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	2WL
	0	0	1	0	1	1	0	1	1	1	1	1	1	1	1	1	4WL
	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	8WL
	0	1	0	0	1	1	1	1	0	1	1	1	1	1	1	1	16WL
	0	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	32WL
	0	1	1	0	1	1	1	1	1	1	0	1	1	1	1	1	10^3
	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	10^4
	1	0	0	0	1	1	1	1	1	1	1	1	0	1	1	1	10^5
	1	0	0	1	1	1	1	1	1	1	1	1	1	0	1	1	10^6
	1	0	1	0	1	1	1	1	1	1	1	1	1	1	0	1	10^7
	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	10^8

NOTE: On the Output, 0 is the Controlling State.

b. Model 3000 Meter Control Interface

The meter control interface for the Model 3000 BERC operates almost exactly like the one used with the model 1210A. The only difference is that the START TEST pulse is positive instead of negative (compares Figures 19 and 20). Referring to Figure 5-14, U6 is the time out one shot U11,3-5,6 is the SYNC RECOVERED flip flop. U11,11-8 is the READY/BUSY flip flop. U13, U14 and U15 are the binary error counters. U16 is the COUNT OVERFLOW flip flop. U42 and U43 buffer the error count data to the DR11-A. U28, U29, and U30 drive the front panel lamps. U5 is the decode for the BLOCK SIZE bits (see Table 9). U22,23 and U37 through U40 are optoisolators. Figure 22 presents the overall meter control interface cabling for the Model 3000.

TABLE 9

TRUTH TABLE FOR 3000 BLOCK SIZE DECODE U5

	INPUT				OUTPUT							
LINE	8	4	2	1	1	2	3	4	5	6	7	
	0	0	0	0	1	1	1	1	1	1	1	10^2
	0	0	0	1	0	1	1	1	1	1	1	10^3
	0	0	1	0	1	0	1	1	1	1	1	10^4
	0	0	1	1	1	1	0	1	1	1	1	10^5
	0	1	0	0	1	1	1	0	1	1	1	10^6
	0	1	0	1	1	1	1	1	0	1	1	10^7
	0	1	1	0	1	1	1	1	1	0	1	10^8
	0	1	1	1	1	1	1	1	1	1	0	10^9

NOTE: On the Output, 0 is the Controlling State.

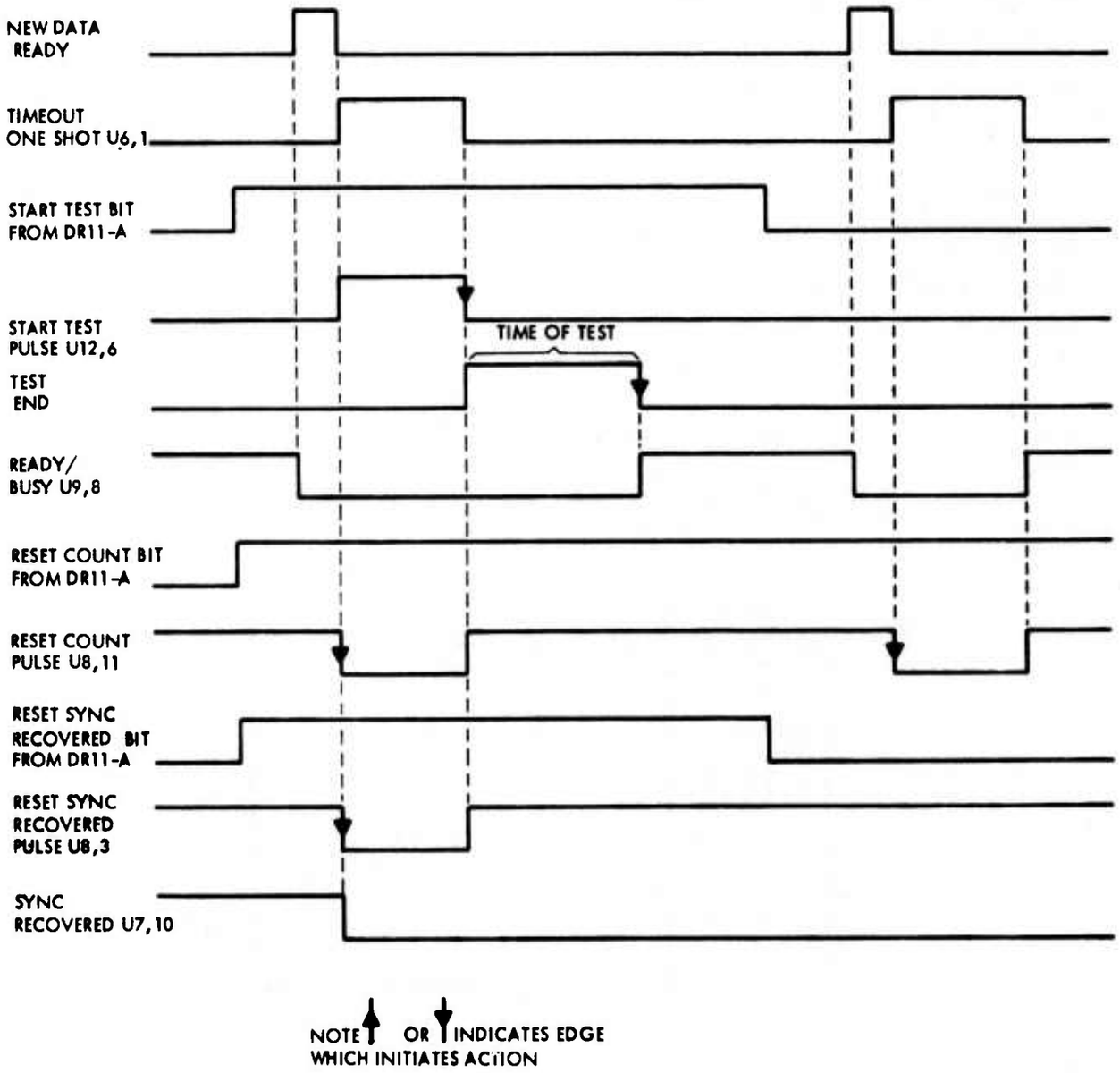
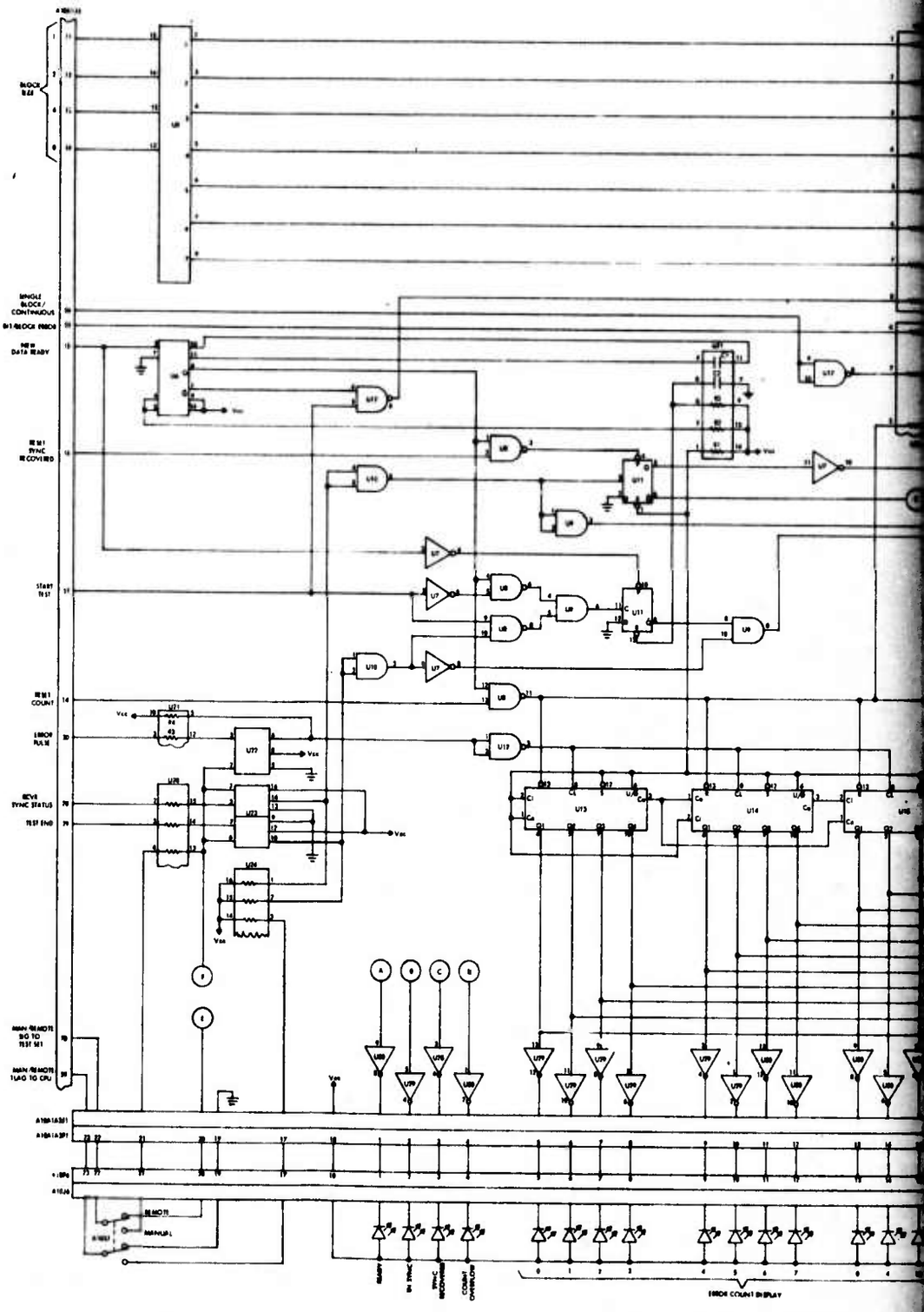


Figure 20. 3000 Meter Control Interface, Timing Diagram



Fig

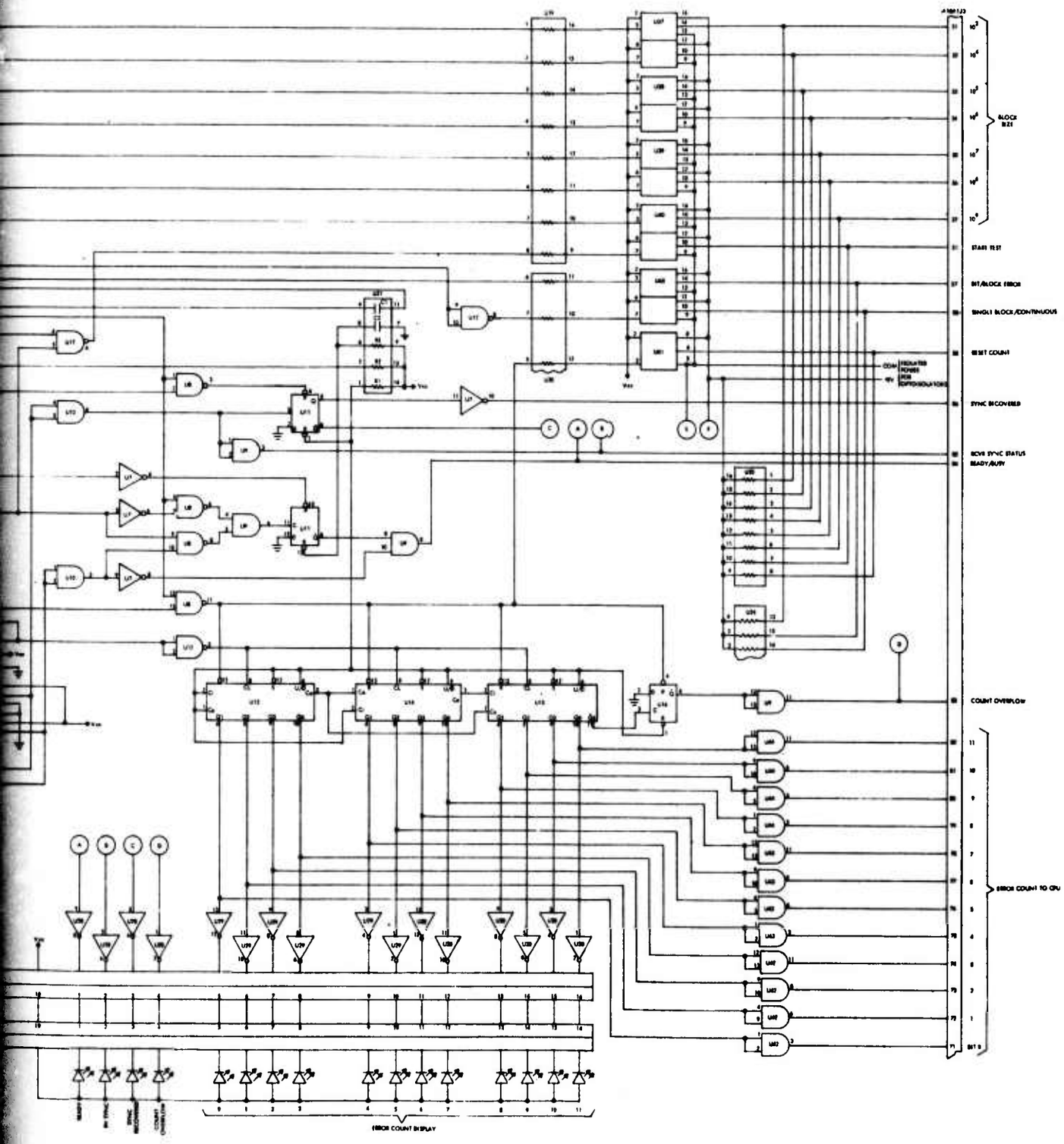
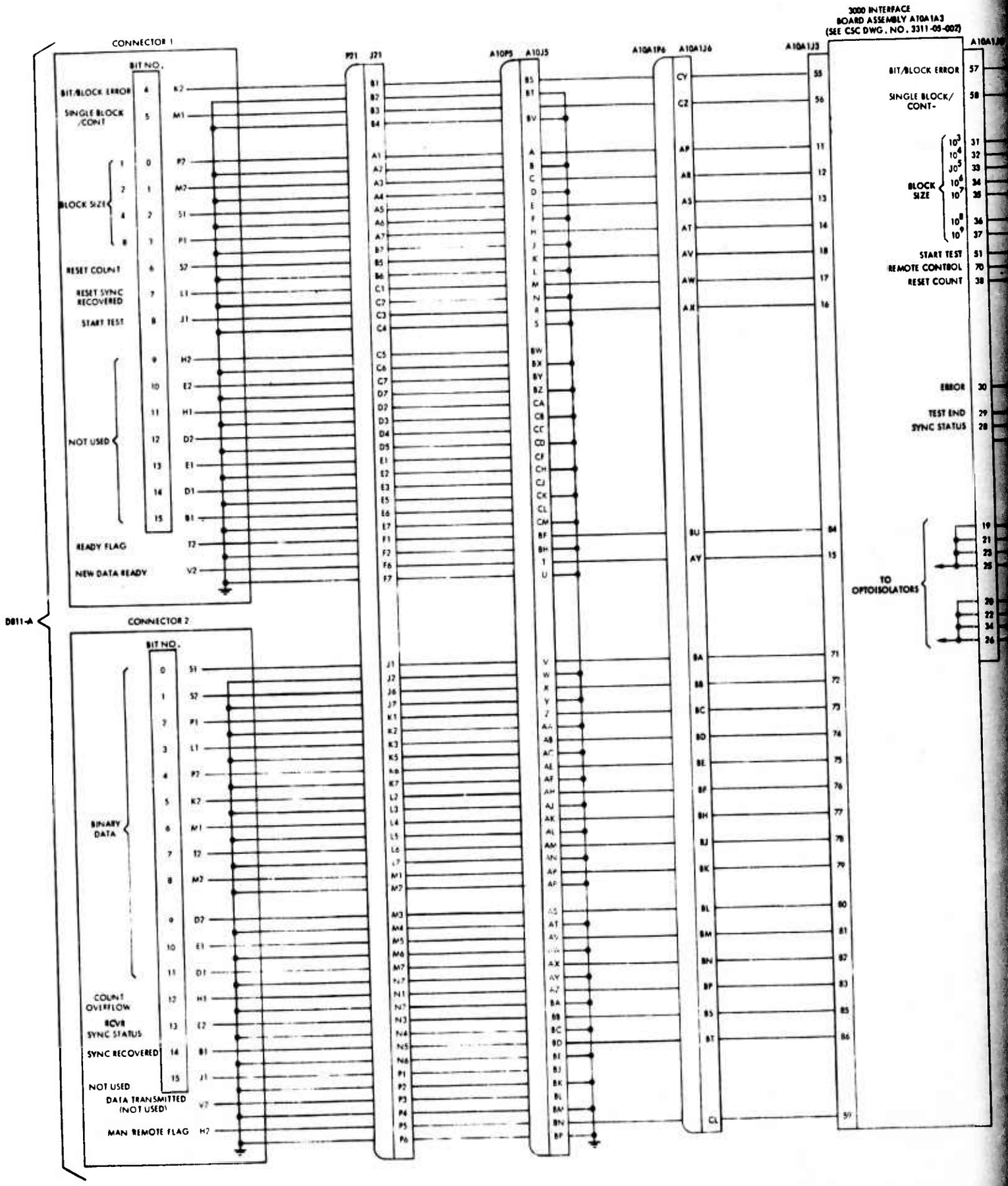


Figure 21. 3000 Interface Board, Unit A10A1A3, Meter Control Section


3000 INTERFACE
BOARD ASSEMBLY A10A1A3
(SEE CSC DWG. NO. 3311-05-002)
CONNECTOR 1
BIT NO.
BIT/BLOCK ERROR
SINGLE BLOCK /CONT
BLOCK SIZE
RESET COUNT
RESET SYNC RECOVERED
START TEST
NOT USED
READY FLAG
NEW DATA READY
CONNECTOR 2
BINARY DATA
COUNT OVERFLOW
RCVR SYNC STATUS
SYNC RECOVERED
NOT USED
DATA TRANSMITTED (NOT USED)
MAN REMOTE FLAG
P21 J21
A10P5 A10J5
A10A1P6 A10A1J6
A10A1J3
A10A1J5
BIT/BLOCK ERROR
SINGLE BLOCK/ CONT-
BLOCK SIZE
START TEST
REMOTE CONTROL
RESET COUNT
ERROR
TEST END
SYNC STATUS
TO OPTOISOLATORS
DB11-A

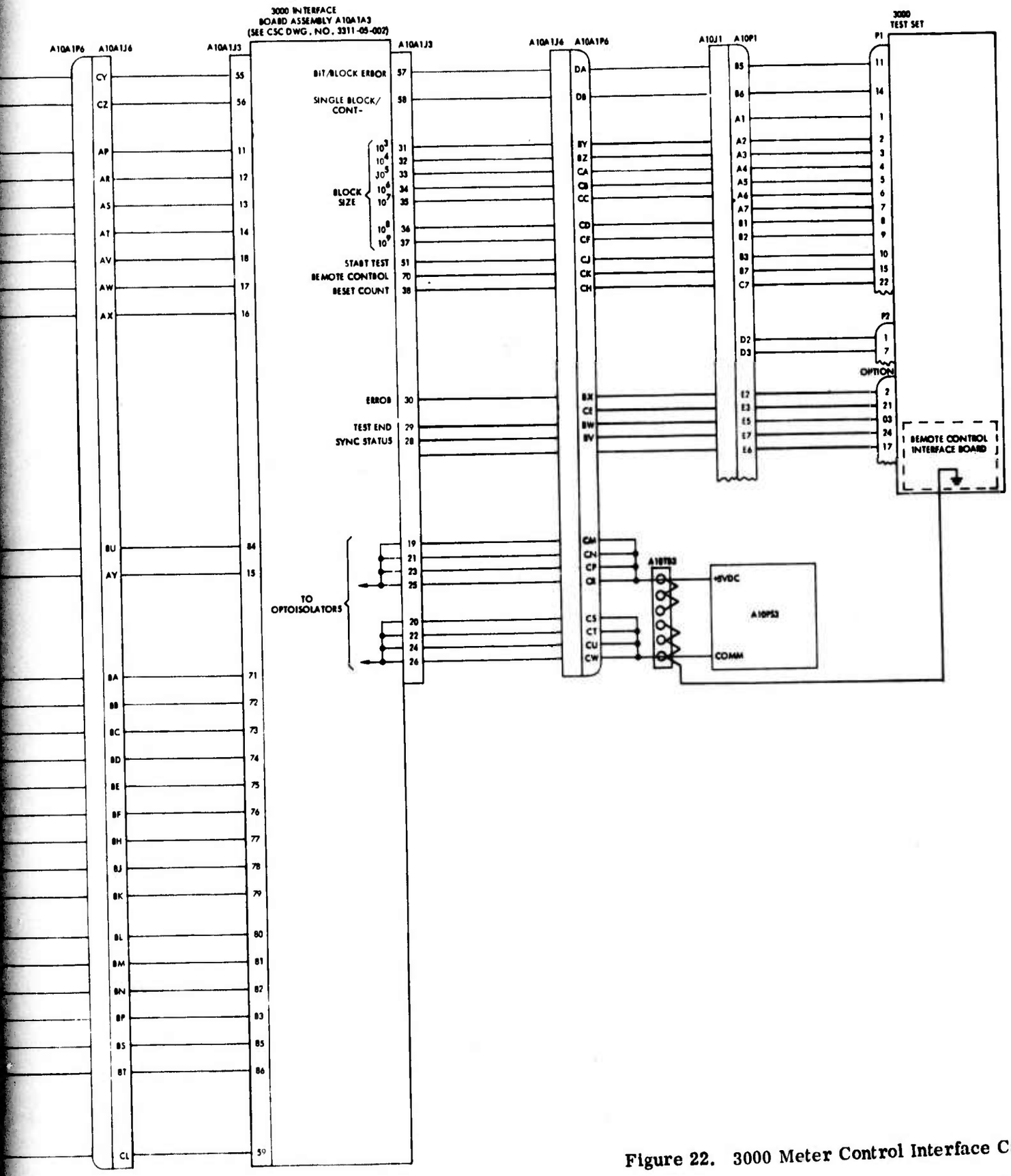


Figure 22. 3000 Meter Control Interface Cabling

SECTION VI

CALIBRATION

Since all the interfaces described herein are digital and no adjustments are provided or necessary, no periodic calibration is required.

This Page Intentionally Left Blank

SECTION VII

TROUBLE SHOOTING AND REPAIR

If either BERC does not respond properly to commands from CSEL, the MAN/REMOTE switch eontrolling it should be ehecked to be sure it is in the REMOTE position. If this is not the trouble, plaee the switeh in the MAN position. The operation of the BERC ean then be tested from its front panel to be sure it is funetioning properly. If the BERC is operating normally, eheek the three power supplies in the interfaee chassis, A10PS1, A10PS2, and A10PS3. They should each be +5 Vdc. Next, with the switeh in the REMOTE position, use the System Exerciser panel in the PDP-11 eomputer to manually set the eontrol bits to the BERC (see Volume II, Seetion 16 of the K-Band Terminal Simulator manual, prepared under Contract F33615-72-C-2187). See Figures 5 and 6 for the address and data format.

To save time it will usually be best to start at the control connectors on the baek of the BERC and probe to see if the eontrol bits correspond to what was set up on the System Exerciser. If not, check the cable connectors at the cabinet I/O panel. If they check, check the ICs on the logie interfaee boards. If the signals at the cabinet I/O connectors do not correspond to the system exerciser panel settings, check the control eables. If tl .bles are all right the trouble is likely in the PDP-11 computer or the DR11-A or UDC-11 interfaces. For these, refer to the appropriate instruction manual.

This Page Intentionally Left Blank

APPENDIX - PARTS LIST

This Page Intentionally Left Blank

Unit BERC Patch Panel

Assembly No. A9

Part	Description	Part No.	Manufacturer
J1 - J6 J8 - J12 J14 - J18	Jack, BNC, Bolkhead	95750	Amphenol
J7, J13	D Receptacle, 25 pin	17 - 10250	Amphenol

Unit BERC Interface Chassis

Assembly No. A10

Part	Description	Part No.	Manufacturer
J1 - J3	Rectangular Receptacle 75 pin	8026-075-000-704	ELCO
J4, J5	Rectangular Receptacle 90 pin	00 8016-90-000-707	ELCO
J6, J7	Circuit Card Header	3432 - 3005	3M Scotchlex
P1 - P3	Rectangular Plug 75 pin	00-8026-075-000-818	ELCO
P4 - P5	Rectangular Plug 90 pin	00-8016-090-000-704	ELCO
P6, P7	Socket Connector	65043-015	Berg
	Pins for J1 - J3 and P1 - P3	60-8216-0313	ELCO
	Pins for J4, J5 and P4, P5	000-60-8017-0313	ELCO
	Pins for P6, P7	47743	Berg
PS1	Power Supply and 5Vdc	LXS-4-5-OVR	Lambda
PS2, PS3	Power Supply Module and 5Vdc	5E50	Acopian
	Socket for PS2, PS3	END-1	Acopian
S1, S2	Switch, Toggle DPDT	MTG-206N	ALCO
TB1	Terminal Block Multi Section	525 (each section)	Buchanan
TB2, TB3	Barrier Terminal Strip	6-141	Cinch
	Front Panel Indicator Lamps	5082-4860	Hewlett Packard

Unit Interface Card Subchassis

Assembly No. A10 A1

Part	Description	Part No.	Manufacturer
J1 - J3	P.C. Board Connector 86 pin	00-6307-086-309-001	ELCO
J4 - J6	Rectangular Receptacle 90 pin	00-8016-90-000-707	ELCO
	Pins for J4 - J6	396-60-8017-0633	ELCO
P4-P6	Rectangular Plug 90 pin	00-8016-090-000-704	ELCO
	Pins for P4 - P6	000-60 8017 03 13	ELCO

Unit 1210A Data Generator Control Interface Board

Assembly No. A10A1A2

Part	Description	Part No.	Manufacturer
U1 - U4	DIP Resistor Array	898-3-R1-5K	Beckman
U5 - U20	OPTO Isolator	5082-5470	Hewlett Packard
U21 - U24	DIP Resistor Array	898-3-R2.2K	Beckman
U25	DIP Resistor Array	34A1024	Allen-Bradley
U26-U28	IC, TTL	SN7442	Texas Instruments
U29	IC, TTL	SN7400	Texas Instruments
U30-U37	IC, TTL	SN75453P	Texas Instruments

Unit 1210A Meter Control Interface Board

Assembly No. A10A1A1

Part	Description	Part No.	Manufacturer
J1	Circuit Card Header	3432-3005	3M Scotchflex
P1	Socket Connector	65043-015	Berg
	Pins for P1	47743	Berg
U1	IC, TTL	SN7400	Texas Instruments
U2	IC, TTL	SN74121	Texas Instruments
U3, U4, U22	DIP Resistor Array	898-3-R1.5K	Beckman
U5, U6	IC, TTL	SN7442	Texas Instruments
U7, U24, U29, U30, U31	IC, TTL	SN7408	Texas Instruments
U8, U35	IC, TTL	SN7400	Texas Instruments
U9	DIP Discrete Component Header	11018-14-3	Scanbe
U10, U28	IC, TTL	SN7474	Texas Instruments
U11, U12, U23	DIP Resistor Array	898-3-R2-2K	Beckman
U13-U21, U36	OPTOISOLATOR	5082-4370	Hewlett Packard
U25, U26, U27	IC, TTL	8284	Signetics
U32, U33, U34	IC, TTL	SN7416	Texas Instruments
C1	Capacitor Ceramic Disc, .01 ufd 50V		CRL
C2	Capacitor Electrolytic 10 ufd 25V		CRL
R1, R2, R3	Resistor, Carbon Composition 1K $\frac{1}{4}$W		Allen Bradley
R4	Resistor, Carbon Composition 10K $\frac{1}{4}$W		Allen Bradley

Unit 3000 Interface Board

Assembly No. A10A1A3

Parts	Description	Part No.	Manufacturer
J1	Circuit Card Header	3432-3005	3M Scotchflex
P1	Socket Connector	65043-015	Berg
U1, U2, U3	IC, TTL	SN75451P	Texas Instruments
U4, U5	IC, TTL	SN7442	Texas Instruments
U6	IC, TTL	SN74121	Texas Instruments
U7	IC, TTL	SN7404	Texas Instruments
U8, U12	IC, TTL	SN7400	Texas Instruments
U9, U10 U42, U43, U44	IC, TTL	SN7408	Texas Instruments
U11, U16	IC, TTL	SN7474	Texas Instruments
U13, U14, U15	IC, TTL	8284	Signetics
U17, U18, U19 U20	DIP Resistor Array	898-3-R1-SK	Beckman
U21	DIP Discrete Component Header	11018-14-3	Scanbe
U22	OPTOISOLATOR	5082-4360	Hewlett Packard
U23, U31-U41	OPTOISOLATOR	5082-4370	Hewlett Packard
U24, U25 U26, U27	DIP Resistor Array	898-3-R2-2K	Beckman
U28, U29, U30	IC, TTL	SN7416	Texas Instruments
R1, R2	Resistor, 10K $\frac{1}{4}$W Carbon Composition		Allen Bradley
R3, R4	Resistor 390Ω$\frac{1}{4}$W Carbon Composition		Allen Bradley
R5	Resistor 10K $\frac{1}{4}$W Carbon Composition		Allen Bradley
C1	Capacitor, Ceramic Disc .01 ufd		CRL
C2	Capacitor, Electrolytic 10 ufd 25V		CRL

☆ U. S. GOVERNMENT PRINTING OFFICE: 1976 — 657-630/491